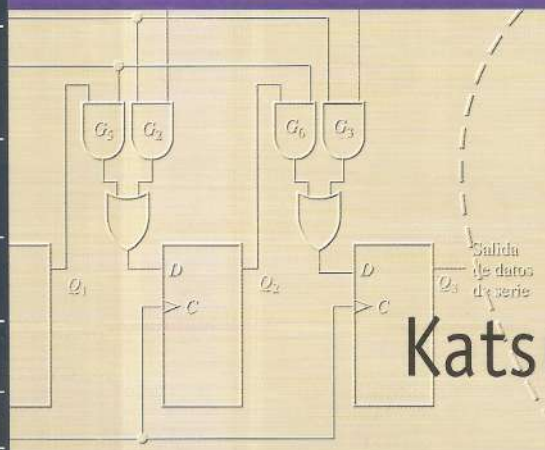
Problemas de

INGENIERÍA DE CONTROL

UTILIZANDO MATLAB®

UN ENFOQUE PRÁCTICO



Katsuhiko Ogata

Prentice Hall

Problemas de Ingeniería de Control utilizando MATLAB®

Katsuhiko Ogata
University of Minnesota

Traducción:
Sebastián Dormido Canto
María Antonia Canto
Departamento de Informática y Automática
Universidad Nacional de Educación a Distancia

PRENTICE HALL
Madrid • México • Santafé de Bogotá • Buenos Aires • Caracas • Lima • Montevideo
San Juan • San José • Santiago • São Paulo • White Plains

Katsuhiko Ogata
Problemas de INGENIERÍA DE CONTROL utilizando Matlab®
PRENTICE HALL IBERIA, Madrid, 1999

Materia: Ingeniería en General 62

Formato 195 x 265 Páginas: 396

Katsuhiko Ogata
Problemas de INGENIERÍA DE CONTROL utilizando Matlab®

No está permitida la reproducción total o parcial de esta obra ni su tratamiento o transmisión por cualquier medio o método sin autorización escrita de la Editorial.

DERECHOS RESERVADOS
© 1999 respecto a la primera edición en español por:
PEARSON EDUCACIÓN, S. A.
C/ Núñez de Balboa, 120
28006 Madrid

Ultima reimpresión, 2000
Traducido de:
Solving Control Engineering problems with MATLAB®
PRENTICE HALL, INC.
Todos los derechos reservados
Simon & Schuster International Group
© MCMXCIV

Editora: Isabel Capella
Diseño de cubierta: Marcelo Spotti
Composición: Sebastián Dormido
Impreso por: GRAFILLES

IMPRESO EN ESPAÑA - PRINTED IN SPAIN

Contenido

PRÓLOGO *vii*

CAPÍTULO 1 MATERIAL INTRODUCTORIO *1*

1-1 Introducción *1*
1-2 Órdenes y funciones matriciales utilizadas en MATLAB *1*
1-3 Estructura del texto *5*

CAPÍTULO 2 FUNDAMENTOS BÁSICOS DE MATLAB *7*

2-1 Introducción *7*
2-2 Introducción de matrices en programas de MATLAB *8*
2-3 Generación de vectores, operaciones matriciales, valores propios y temas relacionados *22*
2-4 Representación gráfica de curvas *31*
2-5 Modelos matemáticos de sistemas lineales *37*

CAPÍTULO 3 ANÁLISIS DE LA RESPUESTA TRANSITORIA DE SISTEMAS CONTINUOS *43*

3-1 Introducción *43*
3-2 Respuesta a una entrada escalón *44*
3-3 Respuesta impulsional *72*
3-4 Respuesta a una entrada en rampa *81*
3-5 Transformación de modelos de sistemas *91*

CAPÍTULO 4 ANÁLISIS DE LA RESPUESTA TRANSITORIA DE SISTEMAS DISCRETOS *99*

4-1 Introducción *99*
4-2 Filtros digitales *103*
4-3 Respuesta a la entrada delta de Kronecker *104*
4-4 Respuesta a una entrada escalón *116*
4-5 Respuesta a una entrada en rampa *126*
4-6 Ejemplos de aplicación *129*
4-7 Respuesta transitoria de sistemas en tiempo discreto definidos en el espacio de estados *139*

CAPÍTULO 5 LUGAR DE LAS RAÍCES 147

5-1 Introducción *147*
5-2 Obtención del lugar de las raíces *149*
5-3 Casos especiales *173*
5-4 Representación de dos o más lugares de las raíces en una misma gráfica *207*
5-5 Lugar de las raíces en el plano z *215*

CAPÍTULO 6 REPRESENTACIÓN GRÁFICA DE LA RESPUESTA DE SISTEMAS DISCRETOS 228

6-1 Introducción *229*
6-2 Fundamentos básicos *229*
6-3 Representación del diagrama de Bode con MATLAB *246*
6-4 Obtención del diagrama de Nyquist con MATLAB *274*
6-5 Ejemplos de aplicación *301*
6-6 Respuesta en frecuencia de sistemas de control en tiempo discreto *326*

REFERENCIAS 353

ÍNDICE 355

Prólogo

MATLAB tiene una excelente colección de órdenes y funciones que son muy útiles para resolver problemas de ingeniería de control. Por ejemplo, la Edición de Estudiante de MATLAB tiene órdenes para:

Respuestas a escalón
Diagramas del lugar de las raíces
Diagramas de respuesta en frecuencia (Diagramas de Bode y Nyquist)
Transformaciones entre modelos en el espacio de estados y modelos en función de transferencia
Valores propios y vectores propios de matrices cuadradas
Conversión de modelos de sistemas continuos a modelos discretos
Diseño de reguladores lineales cuadráticos

entre otras muchas. (Si utiliza la versión profesional de MATLAB, todas estas órdenes están incluidas en el Control System Toolbox.) Como se muestra en este libro, la representación de las curvas de respuesta a escalón, lugares de las raíces, diagramas de Bode y la obtención de soluciones a muchos problemas de control se pueden realizar de una manera fácil con MATLAB.

Las rutinas que aparecen en este libro funcionan con la Edición de Estudiante de MATLAB y con las versiones 3.5 y 4.0 de MATLAB. Las representaciones se hicieron con archivos PostScript creados con la versión 3.5 de MATLAB. La Edición de Estudiante de MATLAB puede presentar las mismas representaciones pero únicamente en una salida en copia dura del volcado de pantalla.

Este libro está destinado a ayudar a los estudiantes de ingeniería y a los ingenieros prácticos en su estudio de MATLAB para resolver los problemas de ingeniería de control. Los problemas discutidos en este libro son básicos en sistemas lineales de control y normalmente son presentados en los cursos introductorios de control. Muchos problemas de ejemplo se han tomado de dos libros anteriores del autor, *Modern Control Engineering*, 2nd edition (Prentice Hall) y *Discrete-Time Control Systems* (Prentice Hall).

En este libro las explicaciones están limitadas a sistemas lineales de control invariantes en el tiempo. Se tratan tanto los sistemas continuos como los muestreados. Todas las señales de entrada consideradas son deterministas.

Una vez que los aspectos teóricos de los problemas de control se han estudiado a través de cursos generales de control, MATLAB puede utilizarse con la ventaja de obtener soluciones numéricas que implican varios tipos de operaciones de vectores y matrices.

Muchos de los programas de MATLAB presentados en este libro se han escrito con comentarios para el usuario, así el lector podrá seguir todos los pasos fácilmente. Por tanto, los lectores que aún no se hayan familiarizado con MATLAB encontrarán este libro muy útil, ya que presenta los detalles de cómo escribir programas de MATLAB para obtener soluciones a los problemas de ingeniería de control. Además este libro proporciona explicaciones detalladas para responder a numerosas preguntas que fueron planteadas en mis cursos de sistemas de control, confío que también respondan a muchas preguntas que se le puedan plantear al lector.

Finalmente, me gustaría agradecer a MathWorks, Inc. por su permiso para utilizar algunos materiales básicos de MATLAB presentados en *Student Edition of MATLAB* (publicado por Prentice Hall) en este libro.

Katsuhiko Ogata

Capítulo 1 Material introductorio

1.1 INTRODUCCIÓN

MATLAB (una abreviatura de MATrix LABoratory) es un sistema basado en el cálculo matricial para desarrollar aplicaciones matemáticas y de ingeniería. Podemos pensar en MATLAB como un clase de lenguaje diseñado únicamente para realizar manipulaciones matriciales. Todas las variables que se manejen en MATLAB son matrices. Esto es, MATLAB tiene solo un tipo de datos, una matriz o un *array* rectangular de números. MATLAB posee un amplio conjunto de rutinas para obtener salidas gráficas.

Este capítulo presenta una introducción a MATLAB. En primer lugar se introducen las órdenes de MATLAB y las funciones matemáticas que se pueden utilizar para resolver problemas de ingeniería de control con MATLAB. A continuación, se presentan los operadores matriciales, los operadores relacionales y lógicos y los caracteres especiales utilizados en MATLAB. El lector debería familiarizarse con todo el material de este capítulo antes de estudiar los programas en MATLAB.

MATLAB posee una ayuda en línea a la que puede llamarse siempre que se desee. La orden help visualizará una lista de funciones y operadores predefinidos para los que hay disponible una ayuda en línea. La orden

help 'nombre de función'

dará información sobre la función específica llamada de su finalidad y forma de uso. La orden

help help

dará información de como utilizar la ayuda en línea.

No se tratan en este libro muchas características importantes y útiles. Para conocer estas particularidades el lector debería consultar la *Edición de Estudiante de MATLAB* y la *Guía de Usuario* que se citan en las referencias.

1.2 ÓRDENES Y FUNCIONES MATRICIALES UTILIZADAS EN MATLAB

En primer lugar se relacionan los diferentes tipos de órdenes y funciones matriciales en MATLAB que se utilizan frecuentemente en la resolución de problemas de ingeniería de control. Después presentaremos de forma breve las conversiones de modelos matemáticos, los operadores matriciales, los operadores relacionales y lógicos y los caracteres especiales empleados en MATLAB.

Órdenes y funciones matriciales en MATLAB que se emplean con frecuencia en la resolución de problemas de ingeniería de control

MATLAB tiene muchas funciones predefinidas que pueden ser llamadas por el usuario para resolver diferentes tipos de problemas.

Conversión del espacio de estado a función de transferencia (ss2tf)
Conversión de función de transferencia a espacio de estado (tf2ss)
Conversión del espacio de estado a ceros-polos (ss2zp)
Conversión de ceros-polos a espacio de estado (zp2ss)
Conversión de función de transferencia a ceros-polos (tf2zp)
Conversión de ceros-polos a función de transferencia (zp2tf)
Conversión de tiempo continuo a tiempo discreto (c2d)

En este libro trataremos con detalle ss2tf, tf2ss y c2d.

Operadores matriciales

Los siguientes signos se utilizan en las operaciones matriciales:

+ Suma
- Resta
* Multiplicación
^ Potencia
' Transpuesta conjugada

Operadores relacionales y lógicos

Los siguientes operadores relacionales y lógicos se utilizan en MATLAB:

< Menor que
<= Menor que o igual a
> Mayor que
>= Mayor que o igual a
== Igual
~= No igual

Observe que '=' se utiliza en una sentencia de asignación, mientras que '==' se emplea en una relación. Los operadores lógicos son:

& AND
| OR
~ NOT

Caracteres especiales

En MATLAB se utilizan los siguientes caracteres especiales:

[ ] Utilizado para formar vectores y matrices
( ) Precedencia de expresión aritmética
, Separa elementos y argumentos de función
; Final de filas, suprime la impresión (véanse los detalles que se dan a continuación)
: Generación de vectores (véanse los detalles que se dan a continuación)
! Ejecución de orden del sistema operativo
% Comentarios (véanse los detalles que se dan a continuación)

Utilización del operador ;

El ; se utiliza para suprimir la impresión. Si el último caracter de una sentencia es un ; se suprime la impresión; la orden se ejecuta pero el resultado no se visualiza. Esto es una característica útil, puesto que la impresión de resultados intermedios puede no necesitarse. También, en la introducción de una matriz el ; se utiliza para indicar el final de una fila excepto de la última.

Utilización del operador :

El operador : juega un papel importante en MATLAB. Este operador se puede utilizar para crear vectores, referenciar submatrices de una matriz dada y especificar los bucles de iteracion *for*. Por ejemplo, j:k es lo mismo que [*j j*+1 ...*k*], A(:,j) es la columna *j*-ésima de A y A(i,:) es la fila *i*-ésima de A.

Línea de programa comenzando con '%'

A lo largo de este libro, muchos programas en MATLAB se escriben con comentarios y observaciones que explican los pasos particulares realizados en el programa. Las líneas de programa en MATLAB que comienzan con '%' son comentarios. La notación '%' es similar a 'REM' en BASIC. Una línea que comienza por % se emplea para almacenar los comentarios del programador y estas observaciones no se ejecutan. Esto es, todo lo que aparece después del signo % en una línea de un programa en MATLAB se ignora. Si los comentarios requieren más de una línea de programa, cada una de ellas debe comenzar con el signo %. (Véase, por ejemplo, el Programa 3-1 en MATLAB.)

1.3 ESTRUCTURA DEL TEXTO

La estructura del libro es la siguiente: este texto está diseñado fundamentalmente para analizar en detalle como presentar los resultados de los análisis de los sistemas de control de forma gráfica.

A continuación se resumen los contenidos de cada capítulo. El Capítulo 1 presenta material introductorio. El Capítulo 2 trata de los fundamentos de las operaciones matriciales con MATLAB.

Los Capítulos 3 y 4 presentan el análisis de la respuesta transistoria de los sistemas de control con MATLAB. El Capítulo 3 se dedica a los sistemas de tiempo continuo y el Capítulo 4 a los sistemas de tiempo discreto. El Capítulo 5 estudia el lugar de las raíces. Analizamos los problemas que pueden surgir al representar el lugar de las raíces con MATLAB. También se incluye el lugar de las raíces para sistemas de control en tiempo discreto. El Capítulo 6 se dedica al análisis de la respuesta en frecuencia de sistemas en lazo cerrado. Tratamos ambos sistemas de control en tiempo continuo y en tiempo discreto. También se presentan en este capítulo problemas de diseño sencillos basados en el diagrama de Bode.

Capítulo 2 Análisis de la respuesta transitoria de sistemas continuos

2.1 INTRODUCCIÓN

Este capítulo trata de las materias que se deben conocer para resolver problemas de ingeniería de control con MATLAB.

Si necesita poner la hora y la fecha

La orden clock da el año, el mes, el día, la hora, los minutos y los segundos. Es decir, clock devuelve un vector fila de seis elementos que contiene la hora y la fecha en formato decimal.

```
clock = [año  mes  día  hora  minutos  segundos]
```

Además, la orden date da la fecha actual.

```
date
ans =
      1-Jan-94
```

Acceso y salida de MATLAB

En la mayoría de los sistemas, una vez que se ha instalado MATLAB, para llamar a MATLAB ejecute la orden MATLAB. Para salir de MATLAB, ejecute la orden exit o la orden quit.

Como se utiliza MATLAB

Normalmente, MATLAB se utiliza en modo de orden dirigida. Cuando las órdenes se introducen en una única línea, MATLAB las procesa inmediatamente y visualiza los resultados. MATLAB, también es capaz de ejecuta secuencias de órdenes que estén almacenadas en filas.

A las órdenes que se hayan escrito, se puede acceder más tarde utilizando la tecla de *flecha hacia arriba*. Es posible encontrar algunas de las últimas órdenes que se hayan introducido y volverlas a llamar en una línea de orden particular.

Variables en MATLAB

Una característica útil de MATLAB es que las variables no necesitan ser dimensionadas antes de ser utilizadas. En MATLAB, las variables se generan de una manera automática una vez que son utilizadas. (Las dimensiones de las variables pueden ser alteradas más tarde si ello fuera necesario.) Estas variables permanecen en memoria hasta que se introduce la orden quit o la orden exit.

Para obtener una lista de las variables en el espacio de trabajo, únicamente escriba la orden who. Después, toda las variables que están actualmente en el espacio de trabajo aparecerán en la pantalla.

La orden clear limpiará todas las variables no permanentes del espacio de trabajo. Si desea limpiar únicamente una variable en particular, por ejemplo 'x', del espacio de trabajo, introduzca la orden clear x.

Como introducir comentarios en un programa en MATLAB

Como se expuso en el Capítulo 1, si se desean introducir comentarios que no van a ser ejecutados, utilice el símbolo % al comienzo de la línea. Es decir, el símbolo % indica que el resto de la línea es un comentario y que debe ser ignorado.

Como guardar variables cuando se sale de MATLAB

Cuando se escribe 'exit o 'quit', todas las variables en MATLAB se pierden. Si se introduce la orden save antes de salir, todas las variables se pueden guardar en un archivo de disco llamado matlab.mat. Cuando se vuelva a entrar en MATLAB, la orden load recuperará el estado inicial del espacio de trabajo.

Estructura del capítulo

La Sección 2-1 proporciona el material introductorio. La Sección 2-2 explica como introducir matrices en programas de MATLAB. La Sección 2-3 presenta la generación de vectores, operaciones con matrices, valores propios y materias que estén relacionadas. La Sección 2-4 muestra como representar las curvas de respuestas, y la Sección 2-5 discute los modelos matemáticos y las transformaciones de sistemas lineales.

2.2 INTRODUCCIÓN DE MATRICES EN PROGRAMAS EN MATLAB

Introducción de señales muestreadas en programas en MATLAB

Los vectores, los cuales son matrices de $1 \times n$ o $n \times 1$, se utilizan de forma normal para guardar señales de datos muestreados en una dimensión, o secuencias. Una manera de introducir una secuencia en MATLAB es introducirla mediante una lista explícita de elementos. Obsérvese que los elementos deben estar separados por espacios en blanco o por comas, como sigue:

```
x = [1  2  3  -4  -5]
```

o

```
x = [1,2,3,-4,-5]
```

Para una mejor lectura es mejor introducir espacios en blanco entre los elementos. Como se muestra, los valores se deben introducir entre corchetes.

La sentencia

```
x = [1  2  3  -4  -5]
```

crea una única secuencia de cinco elementos reales en un vector fila. La secuencia se puede pasar a vector columna transponiéndola. Es decir,

```
y = x'
```

resulta

```
y =

     1
     2
     3
    -4
    -5
```

Como introducir matrices en programas en MATLAB

Una matriz

$$\mathbf{A} = \begin{bmatrix} 1{,}2 & 10 & 15 \\ 3 & 5{,}5 & 2 \\ 4 & 6{,}8 & 7 \end{bmatrix}$$

se puede introducir con un vector fila como sigue:

```
A = [1.2   10   15; 3   5.5   2; 4   6.8   7]
```

Como se muestra, los valores deben ser introducidos entre corchetes. Los elementos de cualquier fila deben estar separados por blancos (o por comas). El final de cada fila, excepto la última, se señala con un punto y coma.

Una matriz grande se puede extender en varias líneas. Por ejemplo, considere la siguiente matriz **B**:

$$\mathbf{B} = \begin{bmatrix} 1{,}5630 & 2{,}4572 & 3{,}1113 & 4{,}1051 \\ 3{,}2211 & 1{,}0000 & 2{,}5000 & 3{,}2501 \\ 1{,}0000 & 2{,}0000 & 0{,}6667 & 0{,}0555 \\ 0{,}2345 & 0{,}9090 & 1{,}0000 & 0{,}3333 \end{bmatrix}$$

Esta matriz se pude extender en cuatro líneas de entrada como sigue:

```
B = [1.5630 2.4572 3.1113 4.1051
     3.2211 1.0000 2.5000 3.2501
     1.0000 2.0000 0.6667 0.0555
     0.2345 0.9090 1.0000 0.3333]
```

Observe que los retornos de carro sustituyen a los puntos y comas.

Otro ejemplo, dada la matriz **C** por

$$\mathbf{C} = \begin{bmatrix} 1 & e^{-0{,}02} \\ \sqrt{2} & 3 \end{bmatrix}$$

se puede introducir como sigue

```
C = [1   exp(-0.02); sqrt(2)   3]
```

Entonces la matriz que se mostrará en la pantalla:

```
C =

    1.0000    0.9802
    1.4142    3.0000
```

Transpuesta y transpuesta conjugada

El apóstrofe (la prima) ' indica la transpuesta conjugada de una matriz. Si la matriz es real, la transpuesta conjugada es únicamente una transpuesta. Una entrada como

```
A = [1  2  3;4  5  6;7  8  9]
```

producirá la siguiente matriz en la pantalla

```
A =

    1    2    3
    4    5    6
    7    8    9
```

También, si se introduce

```
B = A'
```

en la pantalla se verá

```
B =

    1    4    7
    2    5    8
    3    6    9
```

Introducción de matrices complejas

Los números complejos se pueden introducir utilizando la función *i* o *j*. Por ejemplo, un número $1 + j\sqrt{3}$ se puede introducir como

```
x = 1+sqrt(3)*i
```

o

```
x = 1+sqrt(3)*j
```

Este número complejo $1 + j\sqrt{3} = 2\ \exp[(\pi/3)j]$ se puede introducir también como

```
x = 2*exp((pi/3)*j)
```

Es importante observar que, cuando se introducen números complejos como elementos de matrices entre corchetes, se evitan los espacios en blanco. Por ejemplo, 1 + 5*j* se debería introducir como

```
x = 1+5*j
```

Si se ponen espacios en blanco entre el signo +, como

```
x = 1  +  5*j
```

se estarán representando dos números.

Si i y j se utilizan como variables, se debe de generar una nueva unidad compleja como sigue:

```
ii = sqrt(-1)
```

o

```
jj = sqrt(-1)
```

Entonces $-1+j\sqrt{3}$ se debe de introducir como

```
x = -1+sqrt(3)*ii
```

o

```
x = -1+sqrt(3)*jj
```

Introducción de matrices complejas

Si la matriz **X** es una matriz compleja como

$$\mathbf{X} = \begin{bmatrix} 1 & j \\ -j5 & 2 \end{bmatrix}$$

una entrada como

```
X = [1   j;-j*5   2]
```

producirá la siguiente matriz en la pantalla:

```
X =

   1.0000             0+ 1.0000i
      0- 5.0000i   2.0000
```

Observe que

```
Y = X'
```

producirá

```
Y =

   1.0000             0+ 5.0000i
      0- 1.0000i   2.0000
```

que es

$$\mathbf{Y} = \begin{bmatrix} 1 & j5 \\ -j & 2 \end{bmatrix}$$

Puesto que la prima, ', indica la transpuesta conjugada compleja, para una transpuesta no conjugada se utilizará una de las dos siguientes entradas:

```
Y.'        o        conj(Y')
```

Si escribimos

```
Y.'
```

en la pantalla aparecerá

```
ans =

   1.0000             0- 1.0000i
     0+ 5.0000i   2.0000
```

Suma y resta

Las matrices de la misma dimensión se pueden sumar o restar. Considere las siguientes matrices **A** y **B**:

$$\mathbf{A} = \begin{bmatrix} 2 & 3 \\ 4 & 5 \\ 6 & 7 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 1 & 0 \\ 2 & 3 \\ 0 & 4 \end{bmatrix}$$

Si introducimos

```
A = [2   3;4   5;6   7]
```

en la pantalla aparecerá

```
A =

     2     3
     4     5
     6     7
```

Si la matriz **B** se introduce como

```
B = [1   0;2   3;0   4]
```

en la pantalla aparecerá

```
B =

     1     0
     2     3
     0     4
```

Para la suma de dos matrices como **A** + **B**, introducir

```
C = A + B
```

La matriz C aparecerá en la pantalla como

```
C =

   3   3
   6   8
   6   11
```

Si un vector **x** viene dado por

$$\mathbf{x} = \begin{bmatrix} 5 \\ 4 \\ 6 \end{bmatrix}$$

introducimos este vector como

```
x = [5;4;6]
```

La pantalla mostrará el vector columna como

```
x =

   5
   4
   6
```

La siguiente entrada restará 1 de cada elemento del vector **x**.

```
y = x - 1
```

La pantalla mostrará

```
y =

   4
   3
   5
```

Multiplicación de matrices

La multiplicación de matrices se indica por *. Considere

```
x = [1;2;3];   y = [4;5;6];   A = [1   1   2;3   4   0;1   2   5]
```

La entrada

```
x'*y
```

dará

```
ans =
```

```
32
```

Además, la entrada

```
x*y'
```

dará

```
ans =

   4   5   6
   8  10  12
  12  15  18
```

Análogamente, si introducimos

```
y*x'
```

La pantalla mostrará

```
ans =

   4   8  12
   5  10  15
   6  12  18
```

Los productos de matrices y vectores son un caso particular de los productos de matrices. Por ejemplo, una entrada como

```
b = A*x
```

producirá

```
b =

   9
  11
  20
```

Observe que un escalar puede multiplicar, o ser multiplicado por cualquier matriz. Por ejemplo, introduzca

```
5*A
```

dará

```
ans =

   5   5  10
  15  20   0
   5  10  25
```

y una entrada como

```
A*5
```

también dará

```
ans =

     5     5    10
    15    20     0
     5    10    25
```

Matriz exponencial

expm(A) es la matriz exponencial de una matriz A de $n \times n$. Es decir,

$$\text{expm}(\mathbf{A}) = \mathbf{I} + \mathbf{A} + \mathbf{A}^2/2! + \mathbf{A}^3/3! + \ldots$$

Obsérvese que una función trascendental se interpreta como una función matricial si se añade una "m" al nombre de la función, como en expm(A) o sqrtm(A).

Valor absoluto

abs(A) da una matriz que consiste en el valor absoluto de cada elemento de **A**. Si **A** es compleja, abs(A) devuelve el módulo del complejo (magnitud):

```
abs(A) = sqrt(real(A).^2+imag(A).^2)
```

angle(A) devuelve los ángulos de fase en radianes de los elementos de la matriz compleja A. Los ángulos se encuentran entre $-\pi$ y π. Véase el siguiente ejemplo.

```
A=[2+2*i 1+3*i;4+5*i 6-i];
abs(A)

ans =

    2.8284    3.1623
    6.4031    6.0828

angle(A)

ans =

    0.7854    1.2490
    0.8961   -0.1651
```

Magnitud y ángulo de fase de un número complej

La magnitud y el ángulo de fase de un número complejo $z = x + iy = re^{i\theta}$ vienen dados por

```
r = abs(z)
teta = angle(z)
```

y la sentencia

```
z = r*exp(i*teta)
```

recupera el número complejo original z.

Obtención del cuadrado de los elementos de un vector x

Para un vector **x**, x.^2 da el vector del cuadrado de cada elemento. Por ejemplo, para

$$\mathbf{x} = [1 \quad 2 \quad 3]$$

x.^2 se muestra en la siguiente salida de MATLAB:

```
x=[1 2 3];
x.^2

ans =

     1     4     9
```

También, para el vector **y**,

$$\mathbf{y} = [2+5j \quad 3+4j \quad 1-j]$$

y.^2 viene dado por:

```
y=[2+5*i 3+4*i 1-i];
y.^2

ans =

 -21.0000 +20.0000i  -7.0000 +24.0000i        0 - 2.0000i
```

Obtención del cuadrado de los elementos de una matriz A

Para una matriz **A**, A.^2 da una matriz que consiste en el cuadrado de cada elemento. Por ejemplo, para las matrices **A** y **B**, donde

$$\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 1+j & 2-2j \\ 3+4j & 5-j \end{bmatrix}$$

A.^2 y B.^2 vienen dadas como sigue:

```
A=[1 2;3 4];
A.^2

ans =

     1     4
     9    16

B=[1+i 2-2*i;3+4*i 5-i];
B.^2

ans =

        0 + 2.0000i         0 - 8.0000i
  -7.0000 +24.0000i   24.0000 -10.0000i
```

Multiplicación y división de un array

La multiplicación de un array, o elemento por elemento, se indica por '.*'. Si **x** e **y** tienen las mismas dimensiones, entonces

x.*y

indica el array cuyos elementos son únicamente los productos de los elementos individuales de **x** e **y**. Por ejemplo, si

$$\mathbf{x} = [1 \quad 2 \quad 3], \qquad \mathbf{y} = [4 \quad 5 \quad 6]$$

entonces

z = x.*y

resulta

$$\mathbf{z} = [4 \quad 10 \quad 18]$$

Análogamente, si las matrices **A** y **B** tienen las mismas dimensiones, entonces A.*B indica el array cuyos elementos son únicamente los productos de los elementos correspondientes de **A** y **B**. Por ejemplo, si

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 9 & 8 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 4 & 5 & 6 \\ 7 & 6 & 5 \end{bmatrix}$$

Entonces

C = A.*B

resulta

$$\mathbf{C} = \begin{bmatrix} 4 & 10 & 18 \\ 0 & 54 & 40 \end{bmatrix}$$

Las expresiones x./y, x.\y, A./B y A.\B dan los cocientes de los elementos individuales. Así para

$$\mathbf{x} = [1 \quad 2 \quad 3], \qquad \mathbf{y} = [4 \quad 5 \quad 6]$$

la sentencia

```
u = x./y
```

da

$$\mathbf{u} = [0.25 \quad 0.4 \quad 0.5]$$

y la sentencia

```
v = x.\y
```

resulta

$$\mathbf{v} = [4 \quad 2.5 \quad 2]$$

Análogamente, para las matrices **A** y **B**, donde

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 9 & 8 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 4 & 5 & 6 \\ 7 & 6 & 5 \end{bmatrix}$$

la sentencia

```
C = A./B
```

da

$$\mathbf{C} = \begin{bmatrix} 0{,}2500 & 0{,}4000 & 0{,}5000 \\ 0{,}1429 & 1{,}5000 & 1{,}6000 \end{bmatrix}$$

Y la orden

```
D = A.\B
```

da

$$\mathbf{D} = \begin{bmatrix} 4{,}0000 & 2{,}5000 & 2{,}0000 \\ 7{,}0000 & 0{,}6667 & 0{,}6250 \end{bmatrix}$$

Corrección de letras y números mal escritos

Utilice las teclas de flecha del teclado para editar las órdenes erróneas o volver a llamar las líneas de orden previas. Por ejemplo, si se introduce

```
A = (1  1  2]
```

el primer paréntesis debe ser corregido. En lugar de repetir la línea completa, pulse la tecla de *flecha hacia arriba*. Esta línea incorrecta se visualizará otra vez. Utilice la tecla de *flecha hacia la izquierda* y mueva el cursor sobre (y escriba [después pulse la tecla de borrar.

MATLAB es sensible a mayúsculas

Es importante recordar que MATLAB es sensible a mayúsculas en los nombres de órdenes, funciones, y variables. MATLAB distingue entre letras mayúsculas y letras minúsculas. Así, x y X no son la misma variable.

Todos los nombres de función se deben escribir en minúscula, inv(A) invertirá **A**, eig(A) dará los valores propios. Observe, sin embargo, que si la orden casesen off se introduce, MATLAB llegará a ser insensible a las letras mayúsculas y INV(A) es igual a inv(A). Sin embargo para utilizar la orden casesen off, es necesario alguna precaución.

Considere el siguiente ejemplo. Suponga que la matriz **A** viene dada por

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}$$

Al introducir la orden inv(A) se produce la inversa de la matriz **A**. Si introducimos la orden INV(A), la salida de MATLAB muestra un mensaje de error. Si a continuación introducimos la orden casesen off, la salida de MATLAB mostrará un mensaje de error nuevamente. Véase la siguiente salida de MATLAB:

```
A=[0 1 0;0 0 1;-6 -11 -6];
inv(A)

ans =

   -1.8333   -1.0000   -0.1667
    1.0000         0         0
         0    1.0000         0

INV(A)
[[[Undefined function or variable.
Symbol in question MM INV

casesen off
INV(A)
[[[Undefined function or variable.
Symbol in question MM a
```

Para evitar estos mensajes de error, introduzca la sentencia

```
a = [A];
```

antes que se introduzaca la orden casesen off. Después la orden INV(A) producirá la inversa de la matriz **A**. Véase la siguiente salida de MATLAB:

```
A=[0 1 0;0 0 1;-6 -11 -6];
INV(A)
[[[Undefined function or variable.
Symbol in question MM INV

a=[A]
casesen off
INV(A)

ans =

   -1.8333   -1.0000   -0.1667
    1.0000         0         0
         0    1.0000         0
```

Introducción de una sentencia larga que no cabe en una línea

Normalmente una sentencia se termina con un retorno de carro o una tecla de retorno. Si la sentencia que se quiere introducir es demasiado largo para un línea, una marca de tres o más puntos, ..., seguido de un retorno de carro se puede utilizar para indicar que la sentencia continúa en la próxima línea. Un ejemplo es

```
x = 1.234 + 2.345 + 3.456 + 4. 567 + 5.678 + 6.789 ...
    + 7.890 + 8.901 - 9.012;
```

Observe que los espacios en blanco alrededor de los signos =, + y - son opcionales. A menudo estos espacios se ponen para mejorar la lectura.

Introducción de algunas sentencias en una línea

Varias sentencias se pueden situar en una única línea si se separan por comas o puntos y comas. Unos ejemplos son

```
plot(x,y,'o'), text(1,20,'Sistema 1'), text(1,15,'Sistema 2')
```

y

```
plot(x,y,'o'); text(1,20,'Sistema 1'); text(1,15,'Sistema 2')
```

Selección del formato de salida

Todos los cálculos en MATLAB se representan en doble precisión. Sin embargo la salida visualizada se puede fijar a cuatro decimales. Por ejemplo, para el vector

```
x = [1/3   0.00002]
```

MATLAB muestra la siguiente salida

```
x =

   0.3333   0.0000
```

Si al final un elemento de una matriz no es un entero exacto, hay cuatro formatos de salida posibles. La salida visualizada se puede controlar visualizando una de las siguientes órdenes:

```
format short
format long
format short e
format long e
```

Una vez llamado, el formato elegido permanece activo hasta que sea cambiado.

Para el análisis de sistema de control, se suelen utilizar format short y format long. Siempre que se llame a MATLAB y no se introduzca una orden de formato, MATLAB muestra los resultados numéricos en formato corto, como sigue:

```
x=[1/3 0.00002];
x

x =

    0.3333    0.0000

format short; x

x =

    0.3333    0.0000

format long; x

x =

   0.33333333333333   0.00002000000000
```

Si todos los elementos de una matriz o vector son enteros exactos, el format short y el format long producirán el mismo resultado, como se observa a continuación:

```
y=[2 5 40];
y

y =

     2     5    40

format short; y

y =

     2     5    40

format long; y

y =

     2     5    40
```

2.3 GENERACIÓN DE VECTORES, OPERACIONES MATRICIALES, VALORES PROPIOS Y TEMAS RELACIONADOS

Generación de vectores

Los dos puntos, :, es un carácter importante en MATLAB. La sentencia

```
t = 1:5
```

genera un vector fila que contiene los números del 1 al 5 con incremento unidad. Ello produce

```
t =

   1   2   3   4   5
```

Se puede utilizar un incremento distinto a 1. Por ejemplo,

```
t = 1:0.5:3
```

resultará

```
t =

   1.0000   1.5000   2.0000   2.5000   3.0000
```

Se pueden utilizar incrementos negativos. Por ejemplo, la sentencia

```
t = 5:-1:2
```

da

```
t =

   5   4   3   2
```

Otras funciones de generación de vectores incluyen linspace, la cual permite especificar el número de puntos en lugar de los incrementos.

```
x = linspace(-10,10,5)
```

da

```
x =

  -10   -5    0    5   10
```

A continuación considere un vector **x** dado por

```
x = [2  4  6  8  10]
```

Las entradas de un vector individual o una matriz se pueden referenciar con índices entre paréntesis. Por ejemplo, x (3) es tercer elemento de **x** y x ([1 2 3]) son los tres primeros elementos de **x** (es decir, 2,4,6). También, para una matriz **A**, A(3,1) indica la entrada en la tercera fila y primera columna de la matriz **A**.

Normas

La norma de una matriz es un escalar que da alguna medida del tamaño de la matriz. Varias definiciones diferentes normalmente se utilizan varias definiciones diferentes. Una de ellas

$$\text{norm}(\mathbf{A}) = \text{el valor singular mayor de } \mathbf{A}$$

Análogamente, se dispone de varias definiciones para la norma de un vector. Una definición que a menudo se utiliza para la definición de la norma de un vector **x** es

$$\text{norm}(x) = \text{sum}(\text{abs}(x).\text{^}2)\text{^}0.5$$

Véase el siguiente ejemplo:

```
x=[2 3 6];
norm(x)

ans =

     7
```

Valores propios y vectores propios

Si **A** es una matriz de $n \times n$, entonces los n números λ que satisfacen

$$\mathbf{Ax} = \lambda\mathbf{x}$$

son los valores propios de **A**. Se pueden encontrar utilizando la orden

```
eig(A)
```

la cual devuelve los valores propios en un vector columna.

Si **A** es real y simétrica, los valores propios serán reales, pero si **A** no es simétrica los valores propios suelen ser números complejos

Por ejemplo, con

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

La orden

```
eig(A)
```

produce

```
ans =

   0+ 1.0000i
   0- 1.0000i
```

Las funciones en MATLAB pueden tener uno o varios argumentos de salida. Por ejemplo, como se ha visto anteriormente, eig(A) produce un vector columna que está formado por los valores propios de **A**, mientras que una sentencia de doble asignación

```
[X,D] = eig(A)
```

produce los valores propios y vectores propios. Los elementos diagonales de una matriz diagonal **D** son los valores propios, y las columnas de **X** corresponden a los vectores propios tales que

$$\mathbf{AX} = \mathbf{XD}$$

Por ejemplo, si

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}$$

entonces la sentencia

```
[X,D] = eig(A)
```

dará el siguiente resultado:

```
[X,D]=eig(A)

X =

   -0.5774    0.2182   -0.1048
    0.5774   -0.4364    0.3145
   -0.5774    0.8729   -0.9435


D =

   -1.0000         0         0
         0   -2.0000         0
         0         0   -3.0000
```

Los vectores propios son escalados de tal forma que la norma de cada uno de ellos es 1.

Si los valores propios de una matriz son distintos, los vectores propios son siempre independientes y la matriz de vectores propios **X** diagonalizará a la matriz original **A** si se aplica una transformación de semejanza. Sin embargo, si una matriz tiene valores propios repetidos es no diagonalizable a menos que tenga un conjunto completo (independiente) de vectores propios. Si los vectores propios no son independientes, se dice que la matriz original es singular. Incluso si una matriz es singular, la solución de eig satisface la relación **AX = XD**.

Valores propios generalizados y vectores propios generalizados

Si **A** y **B** son matrices cuadradas, entonces la orden

```
eig(A,B)
```

devuelve un vector que contiene los valores propios generalizados al resolver la ecuación

$$\mathbf{Ax} = \lambda \mathbf{Bx}$$

donde λ es un escalar. Los valores de λ que satisfacen la ecuación con los valores propios generalizados, y los valores correspondientes de **x** son los vectores propios generalizados.

Para obtener los vectores propios utilice la orden de doble asignación como sigue:

[X,D] = eig(A,B)

Esto produce una matriz diagonal **D** de valores propios generalizados y una matriz cuadrada **X** cuyas columnas se corresponden con los vectores propios tales que

$$\mathbf{AX} = \mathbf{BXD}$$

Por ejemplo, si

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

eig(A,B) produce

```
eig(A,B)

ans =

  2.1889 + 0.0000i
 -2.1889 - 0.0000i
 -0.4569 - 0.0000i
  0.4569 - 0.0000i
```

y [X,D] = eig(A,B) da

```
[X,D]=eig(A,B)

X =

   0.8463 + 0.0000i  -0.3548 - 0.0000i   0.2433 + 0.0000i   0.1233 - 0.0000i
   0.3866 + 0.0000i   0.1621 + 0.0000i  -0.5326 - 0.0000i   0.2699 - 0.0000i
  -0.2830 - 0.0000i   0.4429 + 0.0000i   0.3899 + 0.0000i   0.7374 - 0.0000i
  -0.2329 - 0.0000i  -0.8073 - 0.0000i  -0.7107 - 0.0000i   0.6068 - 0.0000i


D =

   2.1889 + 0.0000i        0                  0                  0
        0            -2.1889 - 0.0000i        0                  0
        0                  0            -0.4569 - 0.0000i        0
        0                  0                  0            0.4569 - 0.0000i
```

Los vectores propios son escalados de forma que la norma de cada uno es 1.0.

Ecuación característica

Las raíces de la ecuación característica son las mismas que los valores propios de la matriz **A**. La ecuación característica de la matriz **A** se calcula con

```
p = poly(A)
```

Por ejemplo, si la matriz **A** viene dada por

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}$$

la orden poly(A) producirá

```
p=poly(A)

p =

    1.0000    6.0000   11.0000    6.0000
```

Esta es la representación de MATLAB del polinomio

$$s^3 + 6s^2 + 11s + 6 = 0$$

Las raíces de la ecuación característica $p = 0$ se pueden obtener introduciendo la orden r = roots(p):

```
r=roots(p)

r =

   -3.0000
   -2.0000
   -1.0000
```

Las raíces de la ecuación característica pueden recuperar el polimonio original con la orden q = poly(r).

```
q=poly(r)

q =

    1.0000    6.0000   11.0000    6.0000
```

Producto de polinomios

Considere

$$a(s) = s^2 - 20{,}6$$

$$b(s) = s^2 + 19{,}6s + 151{,}2$$

El producto de polinomios es la convolución de los coeficientes. El producto de los polinomios $a(s)$ y $b(s)$ se puede obtener introduciendo la orden c = conv(a,b).

```
a=[1 0 -20.6]; b=[1 19.6 151.2];
c=conv(a,b)

c =

  1.0e+003 *

    0.0010    0.0196    0.1306   -0.4038   -3.1147
```

Esta es la representación en MATLAB del polinomio

$$c(s) = s^4 + 19{,}6s^3 + 130{,}6s^2 - 403{,}8s - 3114{,}7$$

Deconvolución (división de polinomios)

Para dividir el polinomio $c(s)$ entre $a(s)$, utilice la orden de deconvolución [q,r] = deconv(c,a).

```
[q,r]=deconv(c,a)

q =

    1.0000   19.6000  151.2000


r =

     0     0     0     0     0
```

Evaluación polinomial

Si p es un vector cuyos elementos son los coeficientes de un polinomio en potencias descendientes, entonces polyval(p,s) es el valor del polinomio evaluado en s. Por ejemplo, para evaluar el polinomio

$$p(s) = 3s^2 + 2s + 1$$

en $s = 5$, introduzca la orden

```
p = [3  2  1];
polyval(p,5)
```

Se obtiene

```
ans =

    86
```

La orden polyvalm(p,A) evalua el polinomio p de la matriz dada.
Considere la siguiente matriz **J**:

$$\mathbf{J} = \begin{bmatrix} -2+j2\sqrt{3} & 0 & 0 \\ 0 & -2-j2\sqrt{3} & 0 \\ 0 & 0 & -10 \end{bmatrix}$$

La orden poly(J) da el polinomio caractrerístico de **J**.

```
p=poly(J)

p=

    1.0000   14.0000   56.0000  160.0000
```

Esta es la expresión de MATLAB para el polinomio característico de **J**.

$$\text{poly}(\mathbf{J}) = \phi(\mathbf{J}) = \mathbf{J}^3 + 14\mathbf{J}^2 + 56\mathbf{J} + 160\mathbf{I}$$

donde **I** es la matriz identidad. Para la matriz

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}$$

la orden polyvalm(poly(J),A) evalúa el siguiente $\phi(\mathbf{A})$:

$$\phi(\mathbf{A}) = \mathbf{A}^3 + 14\mathbf{A}^2 + 56\mathbf{A} + 160\mathbf{I} = \begin{bmatrix} 154 & 45 & 8 \\ -48 & 66 & -3 \\ 18 & -15 & 84 \end{bmatrix}$$

Véase la siguiente salida de MATLAB.

```
polyvalm(poly(J),A)

ans =

  154.0000   45.0000    8.0000
  -48.0000   66.0000   -3.0000
   18.0000  -15.0000   84.0000
```

Matrices de utilidad

En MATLAB, las funciones

ones(n)
ones(m,n)
ones(A)
zeros

generan matrices especiales. Es decir, ones(n) produce una matriz de unos de $n \times n$. ones(m,n) produce una matriz de unos de $m \times n$. Análogamente, zeros(n) produce una matriz de ceros de $n \times n$, mientras zeros(m,n)

produce una matriz de ceros de $m \times n$. zeros(A) produce una matriz de ceros del mismo tamaño de A, excepto cuando A es un escalar.

Matriz identidad

A menudo necesitamos introducir una matriz identidad **I** en los programas de MATLAB. La sentencia eye(n) da una matriz identidad de $n \times n$. Es decir,

```
eye(5)

ans =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

Matriz diagonal

Si **x** es un vector, la orden diag(x) produce una matriz diagonal con **x** sobre la diagonal. Por ejemplo, para un vector

x = [ones(1,n)]

diag([ones(1,n)]) produce una matriz identidad de $n \times n$ como sigue:

```
diag([ones(1,5])

ans =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

Si **A** es una matriz cuadrada, entonces diag(A) es un vector formado por la diagonal de **A**, y diag(diag(A)) es una matriz diagonal con los elementos de diag(A) sobre la diagonal. Véase la siguiente salida de MATLAB.

```
A=[1 2 3;4 5 6;7 8 9];
diag(A)

ans =

     1
     5
     9

diag(diag(A))

ans =

     1     0     0
     0     5     0
     0     0     9
```

Observe que diag(1:5) da

```
diag(1:5)

ans =

     1     0     0     0     0
     0     2     0     0     0
     0     0     3     0     0
     0     0     0     4     0
     0     0     0     0     5
```

Además, diag(0:4) da

```
diag(0:4)

ans =

     0     0     0     0     0
     0     1     0     0     0
     0     0     2     0     0
     0     0     0     3     0
     0     0     0     0     4
```

Por tanto diag(1:5) - diag(0:4) es una matriz identidad.

```
[diag(1:5)-diag(0:4)]

ans =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

Análogamente, diag(3:7) - diag(2:6) es una matriz identidad.

Es importante observar que diag(0,n) es bastante diferente de diag(0:n). diag(0,n) es una matriz de (n + 1) × (*n* + 1) formada por todos los elementos de ceros. Véase la siguiente salida de MATLAB.

```
diag(0,4)

ans =

     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
```

2.4 REPRESENTACIÓN GRÁFICA DE CURVAS

MATLAB tiene un conjunto extensivo de rutinas para obtener salidas gráficas. La orden plot crea dibujo lineales *x* - *y*. (Los dibujos logarítmicos y polares se crean sustituyendo las palabras loglog, semilogx o polar por plot). Todas estas órdenes se utilizan de la misma manera: únicamente se diferencian en como se escalan los ejes y en como se visualizan los datos.

Gráficas *x-y*

Si **x** e **y** son vectores de la misma longitud, la orden

```
plot(x,y)
```

dibuja los valores de y frente a los valores de x.

Representación de curvas múltiples

Para dibujar varias curvas en un solo gráfico, utilice la orden plot con múltiples argumentos.

```
plot(X1,Y1,X2,Y2, ...,Xn,Yn)
```

Las variables X1,Y1,X2,Y2,...,Xn,Yn son pares de vectores. Se dibuja cada par x-y y se generan múltiples curvas en el gráfico. Los argumentos multiples tienen la ventaja de que permiten visualizar vectores de distinta longitud en un mismo gráfico. Cada par utiliza un tipo de línea distinto.

Para dibujar más de una curva en un único gráfico se puede utilizar también la orden hold. La orden hold congela el gráfico actual e inhibe las acciones de borrado y escalado. Por tanto, las siguientes curvas se dibujarán sobre la curva original. Introduzca nuevamente la orden hold para liberar el gráfico actual.

Inclusión de líneas de rejilla, título de la gráfica, etiqueta en el eje *x* y etiqueta en el eje *y*

Una vez que tiene el gráfico en la pantalla, se pueden dibujar las líneas de rejilla, se puede poner título a la gráfica y los ejes *x*-*y* pueden ser etiquetados. Las órdenes de MATLAB para incluir las líneas de rejilla, el título de la gráfica, la etiqueta en el eje *x* y la etiqueta en el eje *y* son

```
grid      (líneas de rejilla)
```

title (título del gráfico)
xlabel (etiqueta en el eje x)
ylabel (etiqueta en el eje y)

Observe que, una vez que la orden de visualización ha sido ejecutada, las líneas de rejilla, el título del gráfico y las etiquetas en los ejes x e y se pueden introducir sucesivamente en el gráfico introduciendo las órdenes.

Escritura de texto en la pantalla gráfica

Para escribir texto al comienzo del punto (X,Y) sobre la pantalla gráfica, utilice la orden

```
text(X,Y,'texto')
```

Por ejemplo, la declaración

```
text(3,0.45,'sin t')
```

escribirá sin t Horizontalmente empezando en el punto (3,0.45). También, las declaraciones

```
plot(x1,y1,x2,y2), text(x1,y1,'1'), text(x2,y2,'2')
```

marcarán dos curvas para que se puedan distinguir fácilmente. (Véanse los Ejemplos 2-1 y 2-2 para escribir texto sobre pantallas gráficas).

Obtención de una copia dura

Se puede obtener una copia dura del gráfico que se encuentra sobre la pantalla gráfica pulsando la tecla *Print-Scrn*.

EJEMPLO 2-1

Introduzca el siguiente programa de órdenes de MATLAB y muestre el gráfico resultante.

```
t=0:0.05:10;
y=sin(t);
z=cos(t);
plot(t,y,'o',t,z,'x')
grid
title('Gráficas del Seno y del Coseno')
xlabel('Seg')
ylabel('y=seno(t); z=coseno(t)')
text(3,0.45,'sen(t)')
text(0.8,-0.3,'cos(t)')
```

Observe que el vector t es una partición del dominio $0 \leq t \leq 10$ con paso 0.05, mientras que y y z son vectores que dan los valores del seno y del coseno en los puntos de la partición. La Figura 2-1 muestra el gráfico resultante de las curvas del seno y del coseno.

Cuando el gráfico se encuentra en la pantalla, observe que pulsando cualquier tecla MATLAB mostrará la pantalla de órdenes. Con la tecla de *flecha hacia arriba*, introduzca cualquiera de las últimas siete órdenes

(plot, grid, title, xlabel, ylabel, text, text). MATLAB mostrará los gráficos actuales en pantalla. También, si se introduce la orden shg (mostrar gráfico), MATLAB mostrará los gráficos actuales en pantalla.

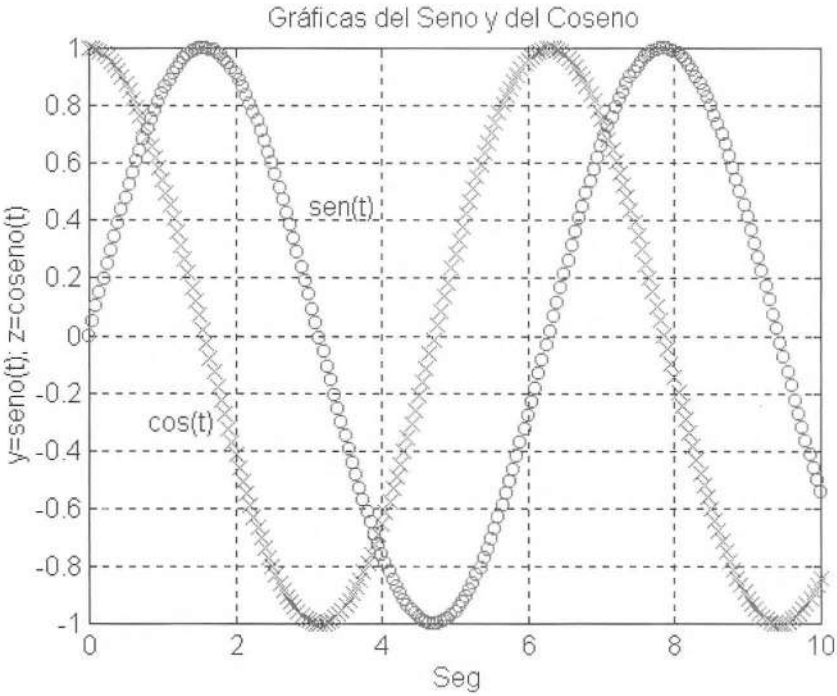


Figura 2-1

EJEMPLO 2-2

Introduzca el siguiente programa de órdenes de MATLAB y muestre el gráfico resultante.

```
t=0:0.25:10;
y=sin(t);
z=cos(t);
plot(t,y,t,z),text(t,y,'y'),text(t,z,'z')
grid
title('Gráficas del Seno y del Coseno')
xlabel('Seg')
ylabel('y=seno(t); z=coseno(t)')
```

La Figura 2-2 muestra el gráfico resultante de las curvas del seno y del coseno.

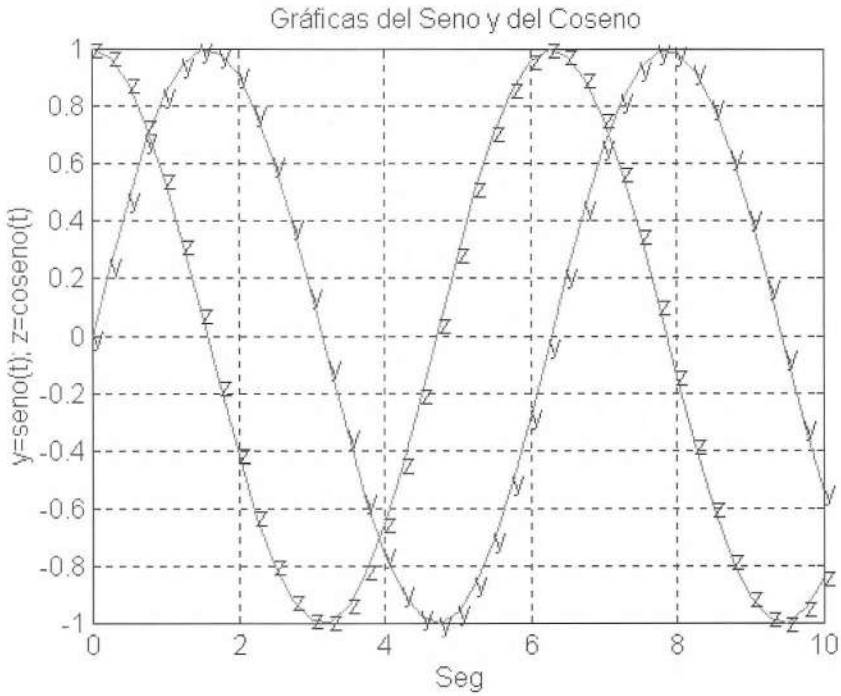


Figura 2-2

EJEMPLO 2-3

Dibujo la gráfica de

$$y = x^2$$

en el intervalo $0 \leq x \leq 3$ con incrementos de 0.1. El programa de MATLAB para este problema del ejemplo es

```
x=0:0.1:3;
y=x.^2;
plot(x,y)
grid
title('Gráfica de y=x^2')
xlabel('x')
ylabel('y')
```

Observe que es necesario que '^2' esté precedido por un punto para asegurarse que se realiza la operación deseada. La Figura 2-3 muestra la gráfica resultante.

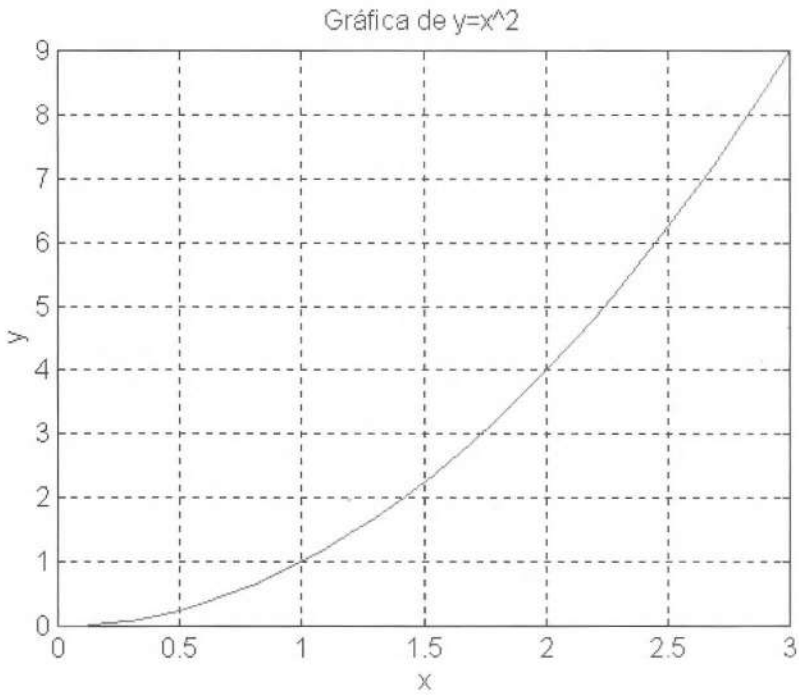


Figura 2-3

Datos imaginarios y complejos

Si **z** es un vector complejo, entonces plot(z) es equivalente a plot(real(z),imag(z))

Diagramas polares

polar(teta,ro) dará un gráfico en coordenadas polares de ángulo teta (en radianes) frente al radio ro. Utilice la orden grid para dibujar las líneas de rejilla del diagrama polar.

Diagramas logarítmicos

log log: un gráfico utilizando escalas $\log_{10}$ - $\log_{10}$
semilogx: un gráfico utilizando escala semilogarítmica; el eje x es $\log_{10}$ y el eje y es lineal.
semilogy: un gráfico utilizando escala semilogarítmica; el eje y es $\log_{10}$ y el eje x es lineal.

Otros tipos de gráficas

bar(x): visualiza un gráfico de barras de los elementos del vectorx; bar no acepta argumentos múltiples.
stairs: parecido a bar, pero pierde las líneas verticales; proporciona un dibujo de escaleras útil para gráficas de señales de sistemas discretos (datos muestreados).

Algoritmos automáticos de representación

En MATLAB el gráfico es escalado automáticamente. Este gráfico permanece como actual hasta que se dibuja otro, en tal caso el gráfico antiguo se elimina y los ejes se reescalan automáticamente. Los algoritmos

automáticos de representación para la respuesta transitoria de curvas, lugar de las raíces, diagramas de Bode, diagramas de Nyquist, etc... son diseñados para trabajar con una gran variedad de sistemas, pero no siempre son perfectos. Así, en algunas ocasiones, puede llegar a ser deseable no hacer caso a la característica de escalado automático de la orden plot y seleccionar manualmente los límites del gráfico.

Escalado manual de ejes

Si se desea dibujar una curva en una región especificada por

v = [x-min x-max y-min y.max]

introduzca la orden axis(v). axis(v), donde v es un vector de cuatro elementos, fija el escalado de ejes a los límites propuestos. Para gráficas logarítmicas, los elementos de v son $\log_{10}$ de los mínimos y los máximos.

Al ejecutar axis(v) se congela el escalado de los ejes actuales para los gráficos siguientes. Escriba nuevamente axis para reanudar el autoescalado.

axis('square') fija una región cuadrada para el gráfico en la pantalla. Con una razón de apariencia cuadrada, una línea con pendiente 1 está exactamente a 45°, y no sesgada por una forma irregular de la pantalla. axis('normal') devuelve la razón de apariencia a normal.

Tipos de gráficas

```
plot(X,Y,'x')
```

dibuja un punto en el gráfico utilizando la marca 'x', mientras

```
plot(X1,Y1,':',X2,Y2,'+')
```

utiliza una línea punteada para la primera curva y el símbolo (+) para la segunda curva. Los tipos de líneas y puntos disponibles son los siguientes:

Tipos de líneas		Tipos de puntos	
solida	-	punto	.
discontinua	--	signo de sumar	+
punteada	:	estrella	*
discontinua-punteada	-.	círculo	o
		marca-x	x

Color

Las declaraciones

```
plot(X,Y,'r')
plot(X,Y,'+g')
```

indican la utilización de una línea roja en el primer gráfico y marcas + de color verde en el segundo. Los colores disponibles son

rojo	r
verde	g

azul	b
blanco	w
invisible	i

2.5 MODELOS MATEMÁTICOS DE SISTEMAS LINEALES

MATLAB tiene órdenes útiles para transformar un modelo matemático de un sistema lineal en otro modelo. Algunas de las transformaciones útiles de sistemas lineales para resolver problemas de ingeniería de control se presentan a continuación.

Función de transferencia a espacio de estados

La orden

```
[A,B,C,D] = tf2ss(num,den)
```

convierte el sistema de función de transferencia

$$\frac{Y(s)}{U(s)} = \frac{\text{num}}{\text{den}} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + D$$

a la representación de espacio de estados

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

$$y = \mathbf{C}\mathbf{x} + Du$$

Espacio de estados a función de transferencia

Si el sistema tiene una entrada y una salida, la orden

```
[num,den] = ss2tf(A,B,C,D)
```

proporciona la función de transferencia $Y(s)/U(s)$.

Si el sistema tiene más de una entrada, utilice la siguiente orden:

```
[num,den] = ss2tf(A,B,C,D,iu)
```

Esta orden convierte el sistema en representación de espacio de estados

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$y = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

a función de transferencia

$$\frac{Y(s)}{U_i(s)} = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$

Observe que el escalar 'iu' es un índice dentro de las entradas del sistema y especifica que entrada se va a utilizar para la respuesta.

Considere, por ejemplo, el siguiente sistema, el cual tiene dos entradas, u_1 y u_2.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Se pueden obtener las dos funciones de transferencia para este sistema. Una relaciona la salida y con la entrada u_1, y la otra relaciona la salida y con la entrada u_2. (Cuando se considera la entrada u_1, se supone que la entrada u_2 es cero, y viceversa). Véase la siguiente salida de MATLAB.

```
A=[0 1;-2 -3];
B=[1 0;0 1];
C=[1 0];
D=[0 0];

[num,den]=ss2tf(A,B,C,D,1)

num =

     0     1     3


den =

     1     3     2


[num,den]=ss2tf(A,B,C,D,2)

num =

     0     0     1


den =

     1     3     2
```

De la salida de MATLAB, tenemos

$$\frac{Y(s)}{U_1(s)} = \frac{s+3}{s^2+3s+2}$$

y

$$\frac{Y(s)}{U_2(s)} = \frac{1}{s^2 + 3s + 2}$$

(Para un estudio más detallado de sistemas con múltiples salidas y entradas, véase la Sección 3-5).

Descomposición en fracciónes parciales de la función de transferencia

Considere la función de transferencia

$$\frac{B(s)}{A(s)} = \frac{\text{num}}{\text{den}} = \frac{b(1)s^n + b(2)s^{n-1} + \ldots + b(n)}{a(1)s^n + a(2)s^{n-1} + \ldots + a(n)}$$

donde $a(1) \neq 0$, pero algún $a(i)$ y $b(j)$ pueden ser ceros.

Los vectores fila num y den especifican los coeficientes del numerador y del denominador de la función de transferencia. es decir,

num = [$b(1)$ $b(2)$ ... $b(n)$]
den = [$a(1)$ $a(2)$ ... $a(n)$]

La orden

[r,p,k] = residue(num,den)

encuentra los residuos, los polos y los términos directos de una descomposición en fracciones parciales del cociente de dos polinomios $B(s)$ y $A(s)$. La descomposición en fracciones parciales de $B(s)/A(s)$ viene dada por

$$\frac{B(s)}{A(s)} = \frac{r(1)}{s - p(1)} + \frac{r(2)}{s - p(2)} + \ldots + \frac{r(n)}{s - p(n)} + k(s)$$

Como ejemplo, considere la siguiente función de transferencia:

$$\frac{B(s)}{A(s)} = \frac{2s^3 + 5s^2 + 3s + 6}{s^3 + 6s^2 + 11s + 6}$$

Para esta función,

num = [2 5 3 6]
den = [1 6 11 6]

La orden

[r,p,k] = residue(num,den)

da el siguiente resultado:

```
[r,p,k]=residue(num,den)

r =

   -6.0000
   -4.0000
    3.0000


p =

   -3.0000
   -2.0000
   -1.0000


k =

     2
```

(Observe que los residuos se devuelven en un vector columna r, la localizacion de los polos en un vector columna p y los términos directos en un vector fila k). Esta es la representación en MATLAB de la siguiente descomposición en fracciones parciales de $B(s)/A(s)$;

$$\frac{B(s)}{A(s)} = \frac{2s^3 + 5s^2 + 3s + 6}{(s+1)(s+2)(s+3)}$$

La orden

$$= \frac{-6}{s+3} + \frac{-4}{s+2} + \frac{3}{s+1} + 2$$

donde r, p, k son dadas en la anterior salida de MATLAB, convierte la descomposición en fracciones parciales al polinomio cociente $B(s)/A(s)$, como sigue:

[num,den] = residue(r,p,k)

```
[num,den]=residue(r,p,k)

num =

    2.0000    5.0000    3.0000    6.0000


den =

    1.0000    6.0000   11.0000    6.0000
```

Conversión de tiempo continuo a tiempo discreto

La orden

$$[G,H] = c2d(A,B,Ts)$$

donde Ts es el periodo de muestreo en segundos, convierte el modelo de espacio de estados de tiempo continuo a discreto, suponiendo un retenedor de orden cero en las entradas. Es decir, con esta orden

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

se convierte en

$$\mathbf{x}(k+1) = \mathbf{G}\mathbf{x}(k) + \mathbf{H}\mathbf{u}(k)$$

Capítulo 3 Análisis de la respuesta transitoria de sistemas continuos

3.1 INTRODUCCIÓN

Respuestas transitorias (tales como respuesta a un salto o entrada escalón, respuesta impulsional y respuesta a una rampa) se utilizan frecuentemente para investigar las características en el dominio temporal de los sistemas de control. Las características de respuesta transitoria tales como tiempo de subida, tiempo de pico, sobreelongación máxima, tiempo de asentamiento y error en estado estacionario se pueden determinar a partir de la respuesta a un salto.

Si se conocen num y den (el numerador y el denominador de la función de transferencia en lazo cerrado,) órdenes tales como

step(num,den) step(num,den,t)

generarán gráficas de respuestas a un salto unitario. (El parámetro t en la orden step es el tiempo especificado por el usuario).

Para un sistema de control definido en el formato del espacio de estados, donde se conocen la matriz de estado **A**, la matriz de control **B**, la matriz de salida **C** y la matriz de transmisión directa **D** de las ecuaciones en el espacio de estado, la orden

step(A,B,C,D,iu,t)

generará gráficas de respuestas a un salto unitario. El vector de tiempos se determina automáticamente cuando el parámetro t no se incluye explícitamente en las órdenes step.

Observe que cuando las órdenes step tienen argumentos en el lado izquierdo, tales como

```
[y,x,t] = step(num,den,t)
[y,x,t] = step(A,B,C,D,iu,t)
[y,x,t] = step(A,B,C,D,iu,t)                    (3-1)
```

ninguna gráfica se muestra en la pantalla. En este caso, es necesario utilizar una orden *plot* para ver las curvas de respuesta. Las matrices x e y contienen la respuesta del estado y de la salida del sistema respectivamente evaluadas en los instantes de tiempo de cálculo t. (y tiene tantas columnas como salidas y una fila para cada elemento de t. x tiene tantas columnas como estados y una fila para cada elemento de t).

Observe en la ecuación (3-1) que el escalar iu es un índice a las entradas del sistema que especifica que entrada se va a utilizar para la respuesta, y t es el tiempo especificado por el usuario. Si el sistema consta de multiples entradas y salidas, la orden *step* tal como viene dada por la ecuación (3-1) produce una serie de gráficas de respuesta a un salto, una por cada combinación de entrada y salida de

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

La respuesta impulsional o la respuesta a una entrada en rampa del sistema de control se puede obtener multiplicando o dividiendo la función de transferencia en lazo cerrado por s y utilizando la orden step. (Véanse las secciones 3-3 y 3-4).

Representación en MATLAB de sistemas lineales

La función de transferencia de un sistema (en el dominio s o z) se representa mediante dos arrays de números. Considérese el sistema

$$G(s) = \frac{2s + 4}{s^3 + 1.3s^2 + 7s + 4}$$

Este sistema se representa como dos arrays cada uno de ellos conteniendo los coeficientes de los polinomios en potencias decrecientes de s tal como sigue:

```
num = [0 0 2 4]
den = [1 1.3 7 4]
```

Observe que hay que rellenar con ceros donde sea necesario.

Es importante darse cuenta que si, por error, introducimos el denominador de esta función de transferencia como

```
den = [1 1,3 7 4]
```

este denominador es completamente diferente del correcto. Debido a la presencia de una coma entre el 1 y el 3, este denominador significa

```
den = [1 1 3 7 4]
```

por tanto, la respuesta del sistema original y la del sistema con el error de mecanografiado son totalmente distintas. Evite siempre esta clase de errores inocentes tales como escribir una coma en lugar de un punto.

Estructura del capítulo

En la sección 3-1 se ha presentado una introducción a la respuesta a un salto. Las secciones 3-2, 3-3 y 3-4 tratan con detalle la respuesta a un salto, la respuesta impulsional y la respuesta a una entrada en rampa respectivamente. La sección 3-5 trata las transformaciones del modelo del sistema desde el espacio de estado a una función de transferencia y viceversa.

3.2 RESPUESTA A UNA ENTRADA ESCALÓN

Comenzaremos esta sección con un ejemplo sencillo.

Obtención de la respuesta a una entrada escalón a partir de la función de transferencia del sistema

Sea el sistema

$$\frac{C(s)}{R(s)} = \frac{25}{s^2 + 4s + 25}$$

Obtener una gráfica de la curva de respuesta a un salto unitario.

El programa 3-1 en MATLAB dará una gráfica de la respuesta a un salto unitario de este sistema. Una gráfica de la curva de respuesta a un salto unitario se muestra en la Figura 3-1.

Programa en MATLAB 3-1

```
%---------------Respuesta a un salto unitario---------------

%******Respuesta a un salto unitario de una función de transferencia******

%***Introduzca el numerador y el denominador de la función de transferencia****

num=[0 0 25];
den=[1 4 25];

%******Introduzca la siguiente orden de respuesta a un salto******

step(num,den)

%******Introduzca la rejilla y el título de la gráfica******

grid
title('Respuesta a un salto unitario de G(s)=25/(s^2+4s+25)')
```

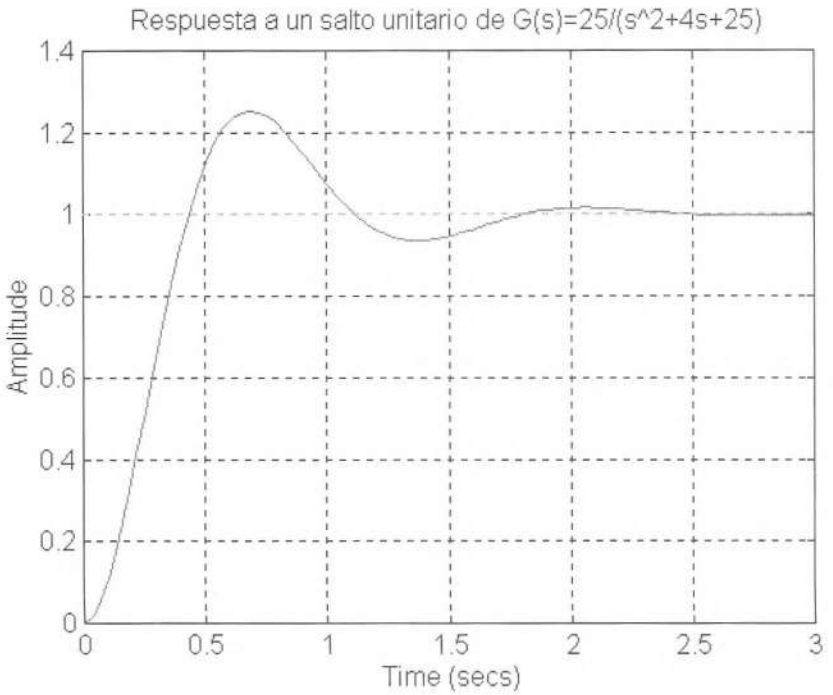


Figura 3-1

EJEMPLO 3-1

Obtener la respuesta a un salto unitario del sistema que se muestra en la Figura 3-2. La función de transferencia en lazo cerrado se puede obtener como sigue:

$$\frac{C(s)}{R(s)} = \frac{6.3223s^2 + 18s + 12.8112}{s^4 + 6s^3 + 11.3223s^2 + 18s + 12.8112}$$

Obtener una gráfica de la curva de respuesta marcada con 'o'.

Para tener una gráfica de la curva de respuesta marcada con 'o', introduzca en el computador el programa 3-2 en MATLAB. La curva de respuesta resultante se muestra en la Figura 3-3.

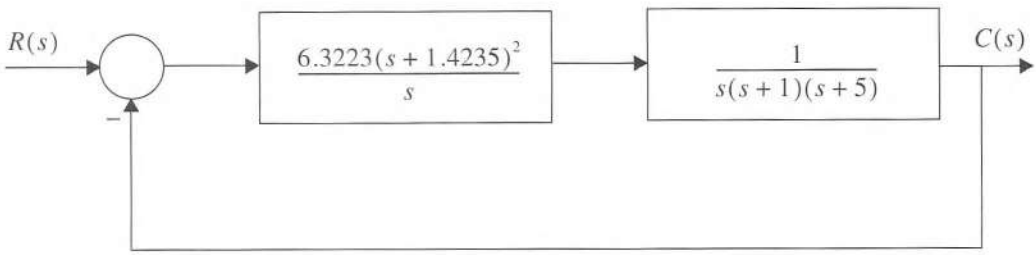


Figura 3-2

Programa en MATLAB 3-2

```
%---------------Respuesta a un salto unitario---------------

%******si se desea respresentar la respuesta a un salto unitario con marcas
%'o', 'x', '--', etc, utilice el programa que se muestra a continuación******

%******Introduzca el numerador y el denominador de la función de transferencia
%en lazo cerrado******

num=[0 0 6.3223 18 12.8112];
den=[1 6 11.3223 18 12.8112];

%******Introduzca la siguiente orden de respuesta a un salto
%y de representación gráfica******

[c,x,t]=step(num,den); plot(t,c,'o')

%**Introduzca la rejilla, título de la gráfica y etiquetas para los ejes x e y**

grid
title('Respuesta a un salto unitario')
xlabel('t seg')
ylabel('salida c')
```

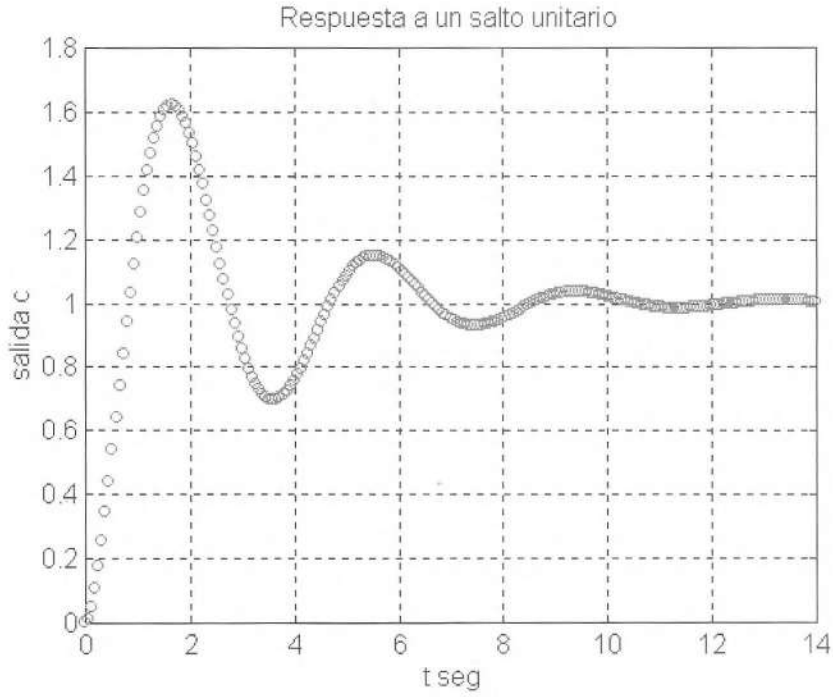


Figura 3-3

EJEMPLO 3-2

Sea el sistema de control que se muestra en la Figura 3-4. En este sistema el controlador proporcional genera el par T que posiciona el elemento de carga, que consiste en un momento de inercia y un rozamiento de tipo viscoso. El par de perturbación al sistema se representa por N.

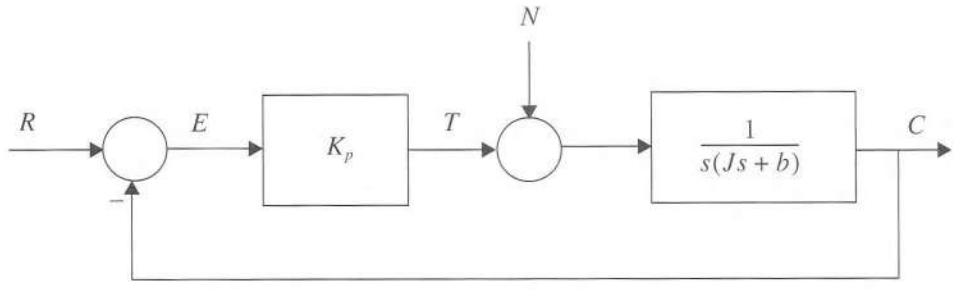


Figura 3-4

Suponiendo que la entrada de referencia es cero o R(s) = 0, la función de transferencia entre C(s) y N(s) viene dada por

$$\frac{C(s)}{N(s)} = \frac{1}{Js^2 + bs + K_p}$$

de donde

$$\frac{E(s)}{N(s)} = -\frac{C(s)}{N(s)} = -\frac{1}{Js^2 + bs + K_p}$$

El error en estado estacionario debido a un salto en el par de perturbación de magnitud T_n viene dado por

$$e_{ss} = \lim_{s \to 0} sE(s) = \lim_{s \to 0} \frac{-s}{Js^2 + bs + K_p} \frac{T_n}{s} = -\frac{T_n}{K_p}$$

En estado estacionario, el controlador proporcional genera el par $-T_n$, que es igual en magnitud pero opuesto en signo al par de perturbación T_n. La salida en estado estacionario debida al salto en el par de perturbación es

$$c_{ss} = -c_{ss} = \frac{T_n}{K_p}$$

El error en estado estacionario se puede reducir aumentando el valor de la ganancia K_p. Aumentando este valor, sin embargo, causará que la respuesta del sistema sea más oscilatoria.

Obtenga las curvas de respuesta a un salto para un pequeño valor de K_p y un gran valor de K_p. Represente las dos curvas de respuesta a un salto en un diagrama.

Consideraremos dos casos:

Caso 1 J = 1, b =.0.5, K_p = 1 (sistema 1)

$$\frac{C(s)}{N(s)} = \frac{1}{s^2 + 0.5s + 1}$$

Caso 2 J = 1, b = 0.5, K_p = 4 (sistema 2)

$$\frac{C(s)}{N(s)} = \frac{1}{s^2 + 0.5s + 4}$$

Observe que para el sistema 1

```
num1 = [0 0 1]
den1 = [1 0.5 1]
```

Para el sistema 2

```
num2 = [0 0 1]
den2 = [1 0.5 4]
```

El programa 3-3 en MATLAB dará las gráficas de dos curvas de respuesta a un salto en un mismo diagrama. En este programa hemos utilizado la notación y1 e y2 para la respuesta. y1 es la respuesta c(t) del Sistema 1, e y2 es la respuesta c(t) del Sistema 2.

Cuando se representan múltiples curvas en un mismo diagrama, se puede utilizar la orden *hold*. Si introducimos la orden hold en el computador, la pantalla mostrará

```
hold

Current plot held
```

Programa en MATLAB 3-3

```
%-------Representación de dos gráficas de respuesta
%a un salto unitario en un mismo diagrama-------

%******Introduzca los numeradores y denominadores de las
%funciones de transferencia******

num1=[0 0 1];
den1=[1 0.5 1];
num2=[0 0 1];
den2=[1 0.5 4];

%******Para representar las dos curvas de respuesta a un salto unitario
%y1 respecto de t e y2 respecto de t en un mismo diagrama introduzca
%las siguientes órdenes******

step(num1,den1);
hold

%Mantiene la grafica actual

step(num2,den2);
grid
title('Respuesta a un salto unitario de dos sistemas')

%******Borrar el mantener las gráficas******

hold

%Libera la gráfica actual
```

Para liberar la gráfica que se mantiene en pantalla, introduzca otra vez la orden hold. De esta forma se libera la gráfica actual. Véanse las siguientes sentencias:

```
hold

Current plot held

hold

Current plot released
```

En la Figura 3-5 se muestran las dos curvas de respuesta a un salto obtenidas.

Obsérvese en la Figura 3-5 que las etiquetas de los ejes x e y se determinan automáticamente. Si se desea etiquetar los ejes x e y de forma diferente, necesitamos modificar la orden step. Por ejemplo, si se desea etiquetar el eje x como 't seg' y el eje y como 'salidas y1 e y2', utilice la orden de respuesta a un salto con argumentos en el lado izquierdo, tal como

$$[y, x, t] = \text{step}(\text{num}, \text{den}, t)$$

Véase el programa 3-4 en MATLAB.

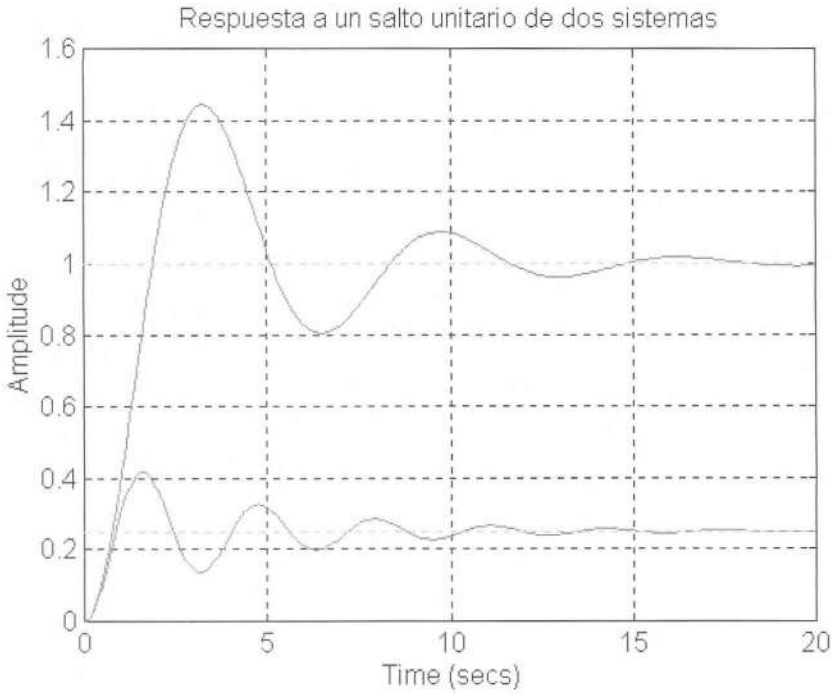


Figura 3-5

Si se desea indicar que las curvas representan respectivamente a los sistemas 1 y 2, podemos introducir la orden *text*. En lo que sigue consideraremos dos casos.

a. Escribir texto en la pantalla gráfica Para escribir texto en la pantalla gráfica, introduzca por ejemplo las siguientes sentencias:

```
text (9, 0.9, 'Sistema 1')
```

y

```
text (9, 0.15, 'Sistema 2')
```

La primera sentencia indica que escriba 'Sistema 1' comenzando en las coordenadas x = 9, y = 0.9. Análogamente la segunda sentencia le dice al computador que escriba 'Sistema 2' empezando en x = 9, y = 0.15. Véase el programa 3-4 en MATLAB. La gráfica resultante se muestra en la Figura 3-6.

En el programa 3-4 en MATLAB, observe que se ha utilizado la orden *plot*, en lugar de emplear la orden *hold*. (Obtenemos el mismo resultado en ambos casos). Para utilizar la orden *plot* con argumentos múltiples, las dimensiones de los vectores y1 e y2 no necesitan ser las mismas. Sin embargo, resulta conveniente que los dos vectores sean de la misma longitud. Por esta razón, indicamos el mismo número de puntos especificando los instantes de cálculo (tal como t = 0: 0.1:20). La orden *step* debe incluir este tiempo t especificado por el usuario. Así, en el programa 3-4 en MATLAB hemos utilizado la siguiente orden *step*:

```
[y, x, t] = step(num, den, t)
```

b. Marcar curvas enteras con texto Para marcar las curvas enteras con texto utilice las siguientes sentencias:

```
text(t,y1,'1')text(t,y2,'2')
```

Programa en MATLAB 3-4

```
%-------Representación de dos gráficas de respuesta a un salto unitario en un
%mismo diagrama-------

%******Introduzca los numeradores y denominadores de las funciones de transfe-
%rencia******

num1=[0 0 1];
den1=[1 0.5 1];
num2=[0 0 1];
den2=[1 0.5 4];

%******Para representar las dos curvas de respuesta a un salto unitario y1
%respecto de t e y2 respecto de t en un mismo diagrama escribiendo los textos
%'Sistema 1' y 'Sistema 2' para distinguir las dos gráficas introduzca las
%siguientes órdenes******

t=0:0.1:20;
[y1,x1,t]=step(num1,den1,t);
[y2,x2,t]=step(num2,den2,t);
plot(t,y1,t,y2), grid, text(9,0.9,'Sistema 1'),text(9,0.15,'Sistema 2')

%******Añadir título a la gráfica y a los ejes x e y******
title('Respuesta a un salto unitario de dos sistemas')
xlabel('t seg'), ylabel('salidas y1 e y2')
```

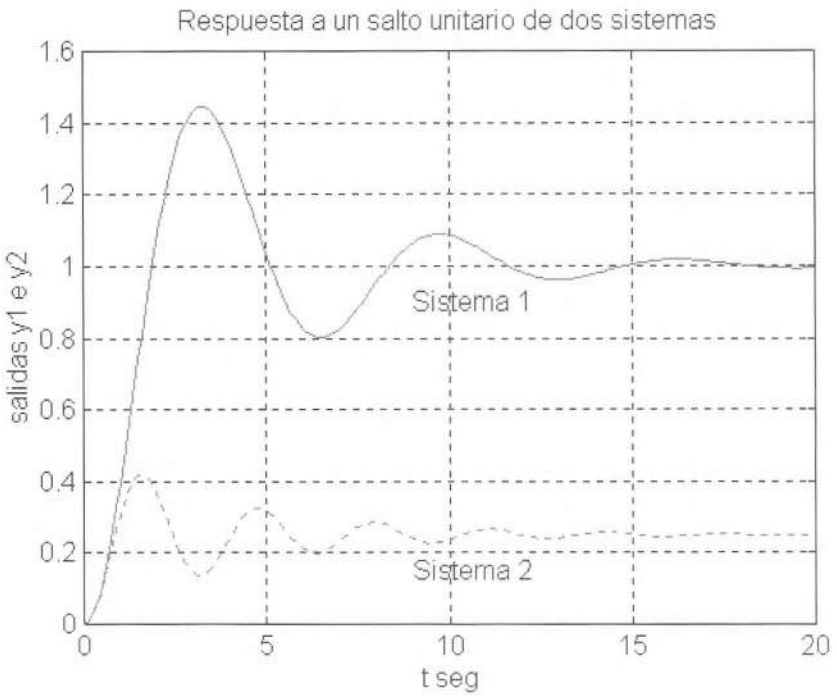


Figura 3-6

Programa en MATLAB 3-3

```
%-------Representación de dos gráficas de respuesta
%a un salto unitario en un mismo diagrama-------

%******Introduzca los numeradores y denominadores de las
%funciones de transferencia******

num1=[0 0 1];
den1=[1 0.5 1];
num2=[0 0 1];
den2=[1 0.5 4];

%******Para representar las dos curvas de respuesta a un salto unitario
%y1 respecto de t e y2 respecto de t en un mismo diagrama introduzca
%las siguientes órdenes******

step(num1,den1);
hold

%Mantiene la grafica actual

step(num2,den2);
grid
title('Respuesta a un salto unitario de dos sistemas')

%******Borrar el mantener las gráficas******

hold

%Libera la gráfica actual
```

Para liberar la gráfica que se mantiene en pantalla, introduzca otra vez la orden hold. De esta forma se libera la gráfica actual. Véanse las siguientes sentencias:

```
hold

Current plot held

hold

Current plot released
```

En la Figura 3-5 se muestran las dos curvas de respuesta a un salto obtenidas.

Obsérvese en la Figura 3-5 que las etiquetas de los ejes x e y se determinan automáticamente. Si se desea etiquetar los ejes x e y de forma diferente, necesitamos modificar la orden step. Por ejemplo, si se desea etiquetar el eje x como 't seg' y el eje y como 'salidas y1 e y2', utilice la orden de respuesta a un salto con argumentos en el lado izquierdo, tal como

$$[y, x, t] = \text{step}(\text{num}, \text{den}, t)$$

Véase el programa 3-4 en MATLAB.

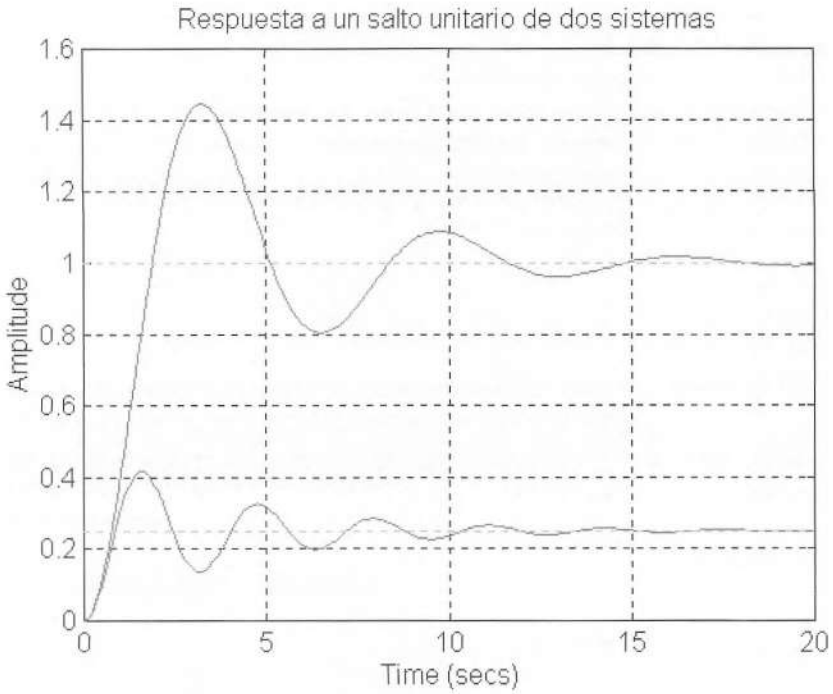


Figura 3-5

Si se desea indicar que las curvas representan respectivamente a los sistemas 1 y 2, podemos introducir la orden *text*. En lo que sigue consideraremos dos casos.

a. Escribir texto en la pantalla gráfica Para escribir texto en la pantalla gráfica, introduzca por ejemplo las siguientes sentencias:

```
text (9, 0.9, 'Sistema 1')
```

y

```
text (9, 0.15, 'Sistema 2')
```

La primera sentencia indica que escriba 'Sistema 1' comenzando en las coordenadas x = 9, y = 0.9. Análogamente la segunda sentencia le dice al computador que escriba 'Sistema 2' empezando en x = 9, y = 0.15. Véase el programa 3-4 en MATLAB. La gráfica resultante se muestra en la Figura 3-6.

En el programa 3-4 en MATLAB, observe que se ha utilizado la orden *plot*, en lugar de emplear la orden *hold*. (Obtenemos el mismo resultado en ambos casos). Para utilizar la orden *plot* con argumentos múltiples, las dimensiones de los vectores y1 e y2 no necesitan ser las mismas. Sin embargo, resulta conveniente que los dos vectores sean de la misma longitud. Por esta razón, indicamos el mismo número de puntos especificando los instantes de cálculo (tal como t = 0: 0.1:20). La orden *step* debe incluir este tiempo t especificado por el usuario. Así, en el programa 3-4 en MATLAB hemos utilizado la siguiente orden *step*:

```
[y, x, t] = step(num, den, t)
```

b. Marcar curvas enteras con texto Para marcar las curvas enteras con texto utilice las siguientes sentencias:

```
text(t,y1,'1')text(t,y2,'2')
```

Programa en MATLAB 3-4

```
%-------Representación de dos gráficas de respuesta a un salto unitario en un
%mismo diagrama-------

%******Introduzca los numeradores y denominadores de las funciones de transfe-
%rencia******

num1=[0 0 1];
den1=[1 0.5 1];
num2=[0 0 1];
den2=[1 0.5 4];

%******Para representar las dos curvas de respuesta a un salto unitario y1
%respecto de t e y2 respecto de t en un mismo diagrama escribiendo los textos
%'Sistema 1' y 'Sistema 2' para distinguir las dos gráficas introduzca las
%siguientes órdenes******

t=0:0.1:20;
[y1,x1,t]=step(num1,den1,t);
[y2,x2,t]=step(num2,den2,t);
plot(t,y1,t,y2), grid, text(9,0.9,'Sistema 1'),text(9,0.15,'Sistema 2')

%******Añadir título a la gráfica y a los ejes x e y******
title('Respuesta a un salto unitario de dos sistemas')
xlabel('t seg'), ylabel('salidas y1 e y2')
```

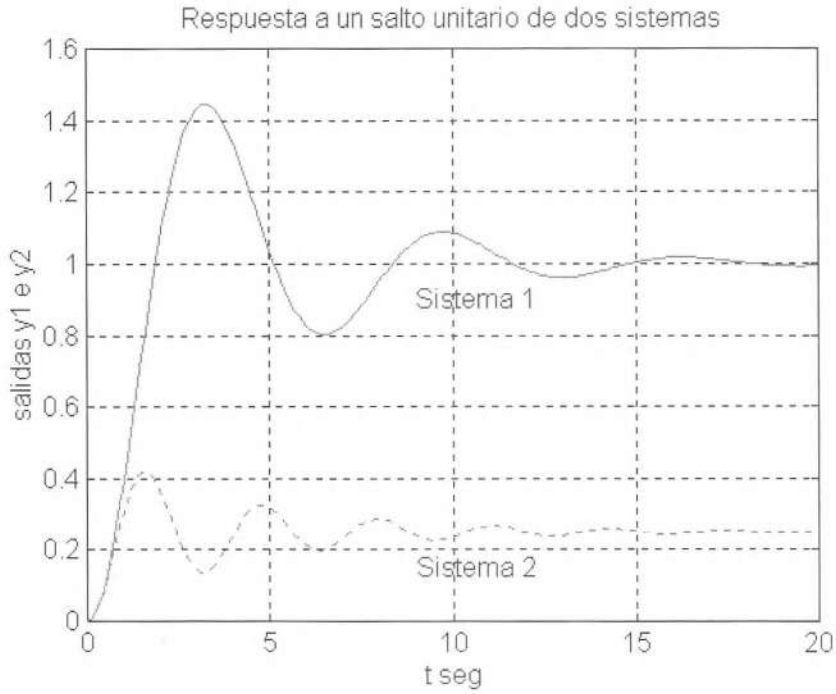


Figura 3-6

Utilizando estas sentencias, las curvas se marcarán con 1 y 2 respectivamente, de forma que se pueden distinguir con facilidad. Para este caso, véase el programa 3-5 en MATLAB. (En este programa utilizamos la orden *plot* con múltiples argumentos.) En la Figura 3-7 se muestra la gráfica resultante.

Programa en MATLAB 3-5

```
%-------Representación de dos gráficas de respuesta
%a un salto unitario en un mismo diagrama-------

%******Introduzca los numeradores y denominadores de las
%funciones de transferencia******

num1=[0 0 1];
den1=[1 0.5 1];
num2=[0 0 1];
den2=[1 0.5 4];

%******Para representar las dos curvas de respuesta a un salto unitario y1
%respecto de t e y2 respecto de t en un mismo diagrama con las gráficas
%marcadas como '1' y '2' respectivamente, introduzca las siguientes órdenes**

t=0:0.4:20;
[y1,x1,t]=step(num1,den1,t);
[y2,x2,t]=step(num2,den2,t);
plot(t,y1,'o',t,y2,'o')
text(t,y1,'1'),text(t,y2,'2')

%******Añadir rejilla, título a la gráfica y a los ejes x e y******
grid
title('Respuesta a un salto unitario de dos sistemas')
xlabel('t seg')
ylabel('salidas y1 e y2')
```

Respuesta a un salto unitario (caso singular)

Para obtener la respuesta a un salto unitario, algunos casos singulares originarán división por cero en los cálculos con MATLAB. Un ejemplo de un caso singular de este tipo se analiza a continuación. Considere la siguiente función de transferencia de un sistema en lazo cerrado

$$\frac{C(s)}{N(s)} = \frac{s}{2s^2 + 2s + 1}$$

donde $N(s)$ es la entrada de perturbación y $C(s)$ es la salida correspondiente. En sistemas de control el efecto de la perturbación debe hacerse tan pequeño como sea posible. (En este sistema para una entrada de perturbación en salto, la salida debida a esta perturbación se hace cero en estado estacionario).

Utilizando estas sentencias, las curvas se marcarán con 1 y 2 respectivamente, de forma que se pueden distinguir con facilidad. Para este caso, véase el programa 3-5 en MATLAB. (En este programa utilizamos la orden *plot* con múltiples argumentos.) En la Figura 3-7 se muestra la gráfica resultante.

Programa en MATLAB 3-5

```
%-------Representación de dos gráficas de respuesta
%a un salto unitario en un mismo diagrama-------

%******Introduzca los numeradores y denominadores de las
%funciones de transferencia******

num1=[0 0 1];
den1=[1 0.5 1];
num2=[0 0 1];
den2=[1 0.5 4];

%******Para representar las dos curvas de respuesta a un salto unitario y1
%respecto de t e y2 respecto de t en un mismo diagrama con las gráficas
%marcadas como '1' y '2' respectivamente, introduzca las siguientes órdenes****

t=0:0.4:20;
[y1,x1,t]=step(num1,den1,t);
[y2,x2,t]=step(num2,den2,t);
plot(t,y1,'o',t,y2,'o')
text(t,y1,'1'),text(t,y2,'2')

%******Añadir rejilla, título a la gráfica y a los ejes x e y******
grid
title('Respuesta a un salto unitario de dos sistemas')
xlabel('t seg')
ylabel('salidas y1 e y2')
```

Respuesta a un salto unitario (caso singular)

Para obtener la respuesta a un salto unitario, algunos casos singulares originarán división por cero en los cálculos con MATLAB. Un ejemplo de un caso singular de este tipo se analiza a continuación. Considere la siguiente función de transferencia de un sistema en lazo cerrado

$$\frac{C(s)}{N(s)} = \frac{s}{2s^2 + 2s + 1}$$

donde $N(s)$ es la entrada de perturbación y $C(s)$ es la salida correspondiente. En sistemas de control el efecto de la perturbación debe hacerse tan pequeño como sea posible. (En este sistema para una entrada de perturbación en salto, la salida debida a esta perturbación se hace cero en estado estacionario).

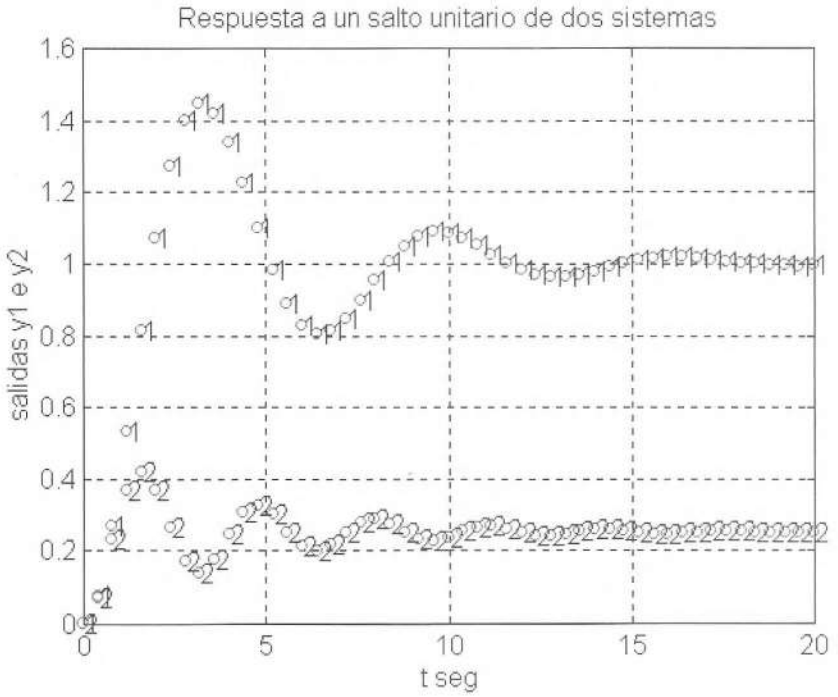


Figura 3-7

Obtengamos la respuesta *c(t)* a la perturbación de un salto unitario a la entrada. Como $N(s) = 1/s$, obtenemos

$$\frac{C(s)}{N(s)} = \frac{s}{2s^2 + 2s + 1}\frac{1}{s} = \frac{1}{2s^2 + 2s + 1}$$

La transformada inversa de Laplace de *C(s)* da

$$c(t) = e^{-0.5t}\sin 0.5t$$

La respuesta *c(t)* se amortigua a cero.

Ahora, obtengamos esta respuesta a un salto con MATLAB. El programa 3-6 en MATLAB se puede utilizar para calcular dicha respuesta. Obsérvese que la orden

```
step(num, den)
```

puede no dar la respuesta a un salto. El mensaje de aviso 'Divide by zero' puede aparecer en la pantalla. Cuando aparece un mensaje de este tipo, introduzca los puntos de tiempo de cálculo explícitamente (tal como t = 0:0.1:12) e introduzca el tiempo t especificado por el usuario en la orden *step* como sigue:

```
step(num, den, t)
```

Así se puede obtener la respuesta *c(t)*. En la Figura 3-8 se muestra la gráfica generada por el programa 3-6 en MATLAB.

Una forma alternativa de evitar este tipo de dificultad (división por cero) es cambiar ligeramente num y/o den. Por ejemplo, en este problema si cambiamos el denominador polinomial de $2s^2 + 2s + 1$ a $2s^2 + 2s + 0.999$ podemos eliminar esta dificultad.

Programa en MATLAB 3-6

```
%-------Respuesta a un salto unitario-------

%******Introduzca el numerador y el denominador de la función de transferen-
%cia******

num=[0 1 0];
den=[2 2 1];

%******Introduzca la orden de respuesta a un salto******

t=0:0.1:12;
step(num,den,t);

%******Añadir rejilla y el título de la gráfica******
grid
title('Respuesta a un salto unitario')
```

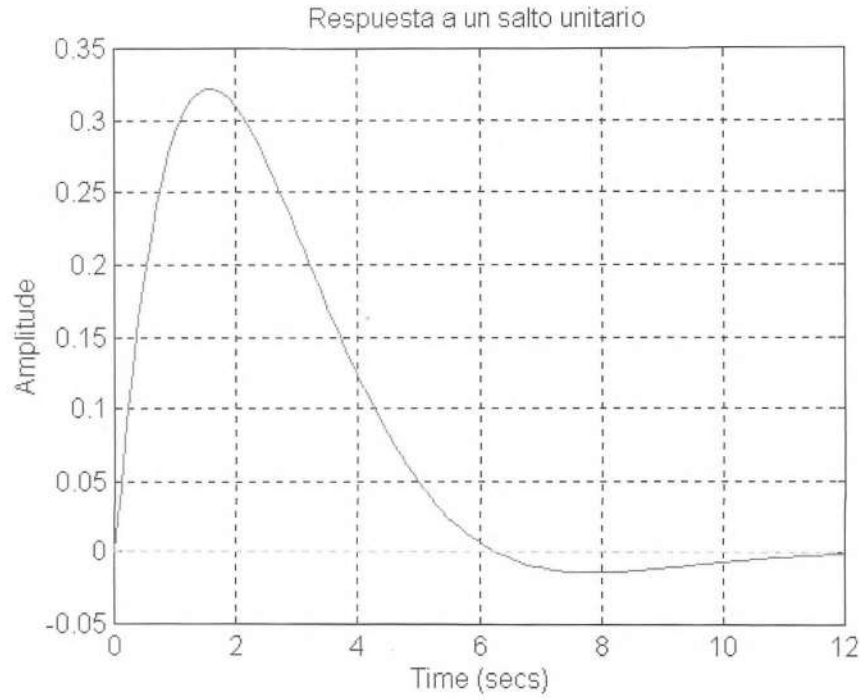


Figura 3-8

EJEMPLO 3-3

Considere el sistema que se muestra en la Figura 3-9. Obtenga las respuestas del sistema a una entrada de referencia y a una entrada de perturbación tipo salto unitario.

Respuesta a una entrada de referencia tipo salto unitario. La función de transferencia en lazo cerrado para el sistema, suponiendo $U_d(s) = 0$, se obtiene como sigue:

$$\frac{C(s)}{N(s)} = \frac{\dfrac{10.4(s^2 + 4.5192s + 15.385)}{s} \dfrac{1}{s^2 + 3.6s + 9}}{1 + \dfrac{10.4(s^2 + 4.5192s + 15.385)}{s(s^2 + 3.6s + 9)}} = \frac{10.4s^2 + 47s + 160}{s^3 + 14s^2 + 56s + 160}$$

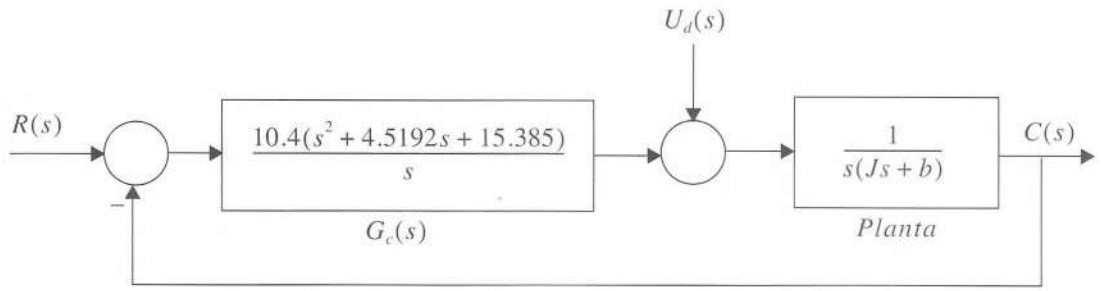


Figura 3-9

Por tanto, la respuesta a un salto unitario de este sistema, cuando no hay perturbación a la entrada se puede obtener introduciendo en el computador el programa 3-7 en MATLAB. En la Figura 3-10 se muestra una gráfica de esta respuesta.

Respuesta a una entrada de perturbación tipo salto unitario. La respuesta en este caso se puede obtener suponiendo que la entrada de referencia es nula. La función de transferencia en lazo cerrado entre C(s) y $U_d(s)$ se obtiene como sigue:

$$\frac{C(s)}{U_d(s)} = \frac{\dfrac{1}{s^2 + 3.6s + 9}}{1 + \dfrac{10.4(s^2 + 4.5192s + 15.385)}{s} \dfrac{1}{s^2 + 3.6s + 9}} = \frac{s}{s^3 + 14s^2 + 56s + 160}$$

Programa en MATLAB 3-7

```
%-------Respuesta a un salto unitario en la entrada de referencia-------

num=[0 10.4 47 160];
den=[1 14 56 160];
v=[0 5 0 1.4];
axis(v);
t=0:0.05:5;
step(num,den,t);
grid
title('Respuesta a un salto unitario en la entrada de referencia')
```

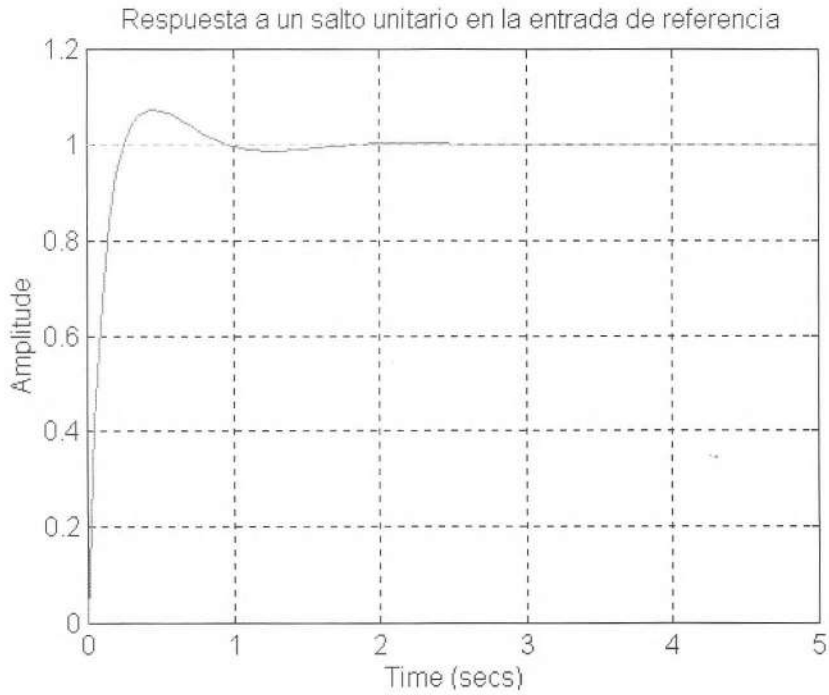


Figura 3-10

El programa 3-8 en MATLAB se puede introducir en el computador para obener una gráfica de la respuesta a una perturbación de entrada tipo salto unitario.

En la Figura 3-11 se muestra una gráfica de esta respuesta.

Programa en MATLAB 3-8

```
%-------Respuesta a un salto unitario en la entrada de perturbación-------

num=[0 0 1 0];
den=[1 14 56 160];
v=[0 10 -0.04 0.04];
axis(v);
t=0:0.1:10;
step(num,den,t);
grid
title('Respuesta a un salto unitario en la entrada de perturbación')
```

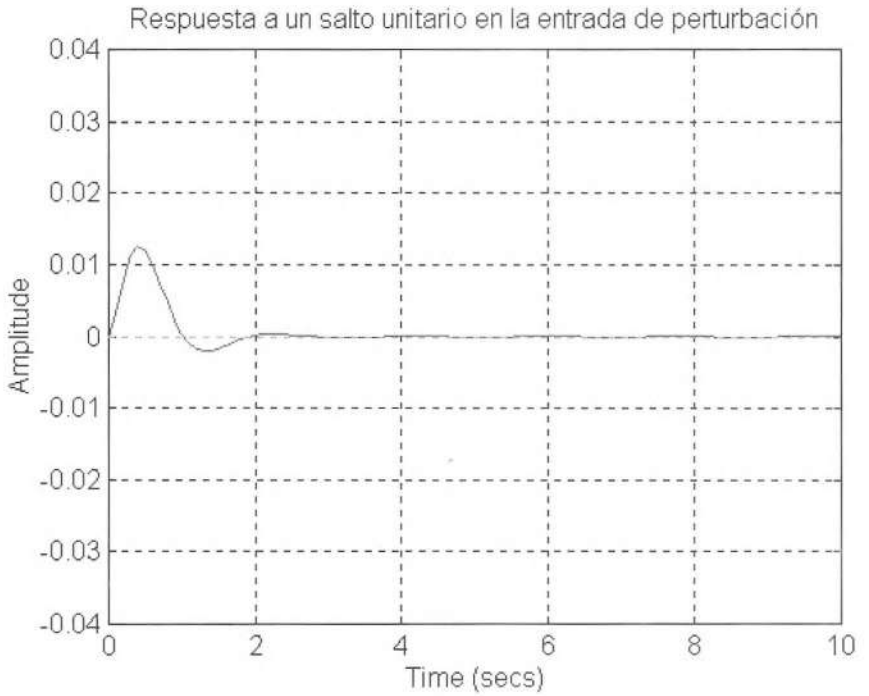


Figura 3-11

EJEMPLO 3-4

En este ejemplo, consideraremos un sistema sujeto solamente a una condición inicial.

Sea el sistema mecánico que se muestra en la Figura 3-12, donde m = 1 kg, b = 3 N-s/m, y k = 2 N/m. Suponga que en t = 0 se tira de la masa m hacia abajo de forma que x(0) = 0.1 m y x'(0) = 0.05 m/s. Obtenga el movimiento de la masa sujeta a la condición inicial. (Suponga que no existe función forzante externa.)

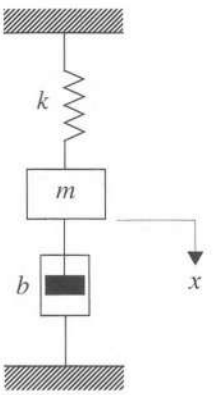


Figura 3-12

La ecuación del sistema es

$$m\ddot{x} + b\dot{x} + kx = 0$$

con las condiciones iniciales x(0) = 0.1 m y x'(0) = 0.05 m/s. La transformada de Laplace de la ecuación del sistema da

$$m[s^2X(s) - sx(0) - \dot{x}(0)] + b[sX(s) - x(0)] + kX(s) = 0$$

o

$$(ms^2 + bs + k)X(s) = mx(0)s + m\dot{x}(0) + bx(0)$$

Resolviendo esta última ecuación para $X(s)$ y sustituyendo los valores numéricos dados, obtenemos

$$X(s) = \frac{mx(0)s + m\dot{x}(0) + bx(0)}{ms^2 + bs + k} = \frac{0.1s + 0.35}{s^2 + 3s + 2}$$

Esta ecuación se puede escribir como

$$X(s) = \frac{0.1s^2 + 0.35s}{s^2 + 3s + 2}\frac{1}{s}$$

Por tanto, el movimiento de la masa m se puede obtener como la respuesta a un salto unitario del siguiente sistema

$$X(s) = \frac{0.1s^2 + 0.35s}{s^2 + 3s + 2}$$

El programa 3-9 en MATLAB dará una gráfica del movimiento de la masa. En la Figura 3-13 se muestra la gráfica.

Programa en MATLAB 3-9

```
%-------Respuesta a condiciones iniciales-------

num=[0.1 0.35 0];
den=[1 3 2];

%*****Introduzca la siguiente orden de respuesta a un salto unitario*****

step(num,den);

%*****Introduzca la rejilla y el título de la gráfica*****

grid
title('Respuesta del sistema masa-resorte-amortiguador a condiciones
iniciales')
```

Figura 3-13

Obtención de la respuesta a un salto unitario del sistema en la forma de espacio de estados

Calcular la respuesta a un salto unitario del siguiente sistema

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$
$$y = \mathbf{C}\mathbf{x} + Du$$

donde

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -100 & -80 & -32 & -8 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 5 \\ 60 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \quad D = 0$$

El programa 3-10 en MATLAB producirá una gráfica de la curva de respuesta a un salto unitario. En la Figura 3-14 se muestra la curva resultante.

Programa en MATLAB 3-10

```
%-------Respuesta a un salto unitario-------

%*****Introduzca las matrices A,B,C y D de las ecuaciones en el
%espacio de estados*****

A=[0 1 0 0;0 0 1 0;0 0 0 1;-100 -80 -32 -8];
B=[0;0;5;60];
C=[1 0 0 0];
D=[0];

%*****Para obtener la respuesta a un salto unitario de y respecto a t,
%introduzca la siguiente orden*****

step(num,den);

step(A,B,C,D)
grid
title('Respuesta a un salto unitario')
```

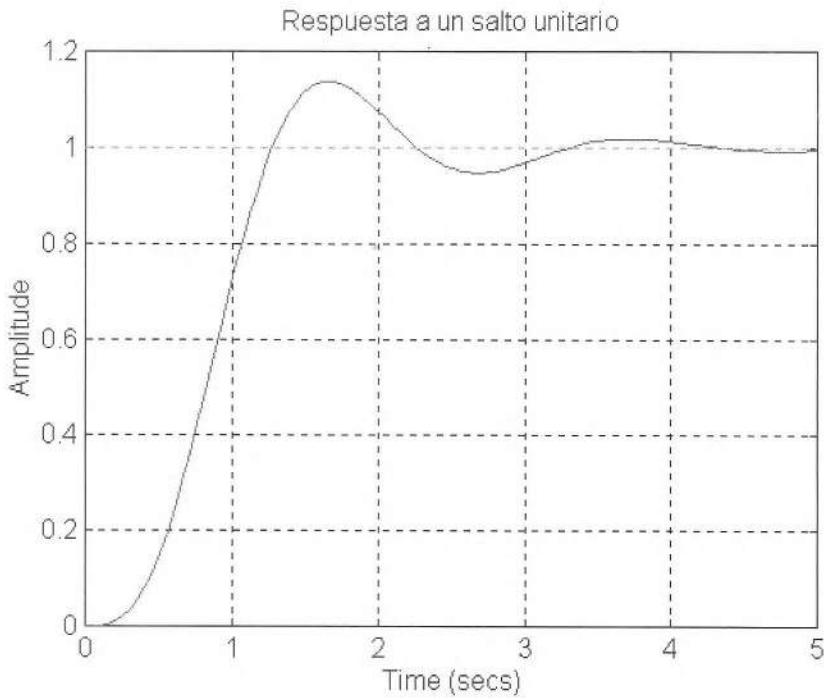


Figura 3-14

EJEMPLO 3-5

Obtener la respuesta a un salto unitario del sistema descrito por las siguientes ecuaciones en el espacio de estado

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5.008 & -25.1026 & -50325 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 25.04 \\ -121.005 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

El programa 3-11 en MATLAB dará una gráfica de la respuesta a un salto unitario del sistema. Una gráfica de la salida $y(t)$ respecto de t se muestra en la Figura 3-15. Es importante observar que si damos la orden

plot(y)

en lugar de plot(t,y), entonces y se representa en los puntos de cálculo (280 puntos = 10 segundos en este ejemplo). Véase la Figura 3-16 para la gráfica de y respecto de los puntos de cálculo. Para representar y(t) respecto de t, es necesario introducir la orden plot(t,y).

Para representar las curvas x1 respecto de t, x2 respecto de t y x3 respecto de t en un mismo diagrama, simplemente introduzca la orden plot(t,x). Para representar x2 respecto de t y x3 respecto de t en diagramas separados, observe que

$$x2 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \times x'$$

$$x3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \times x'$$

Programa en MATLAB 3-11

```
%-------Respuesta a un salto unitario-------

%*****Introduzca las matrices A,B,C y D de las ecuaciones en el
%espacio de estados*****

A=[0 1 0 ;0 0 1 ;-5.008 -25.1026 -5.0325];
B=[0;25.04;-121.005];
C=[1 0 0];
D=[0];

%*****Introduzca la siguiente orden de respuesta a un salto unitario*****

[y,x,t]=step(A,B,C,D);
plot(t,y), grid
title('Respuesta a un salto unitario'), xlabel('t seg'), ylabel('Salida y')
```

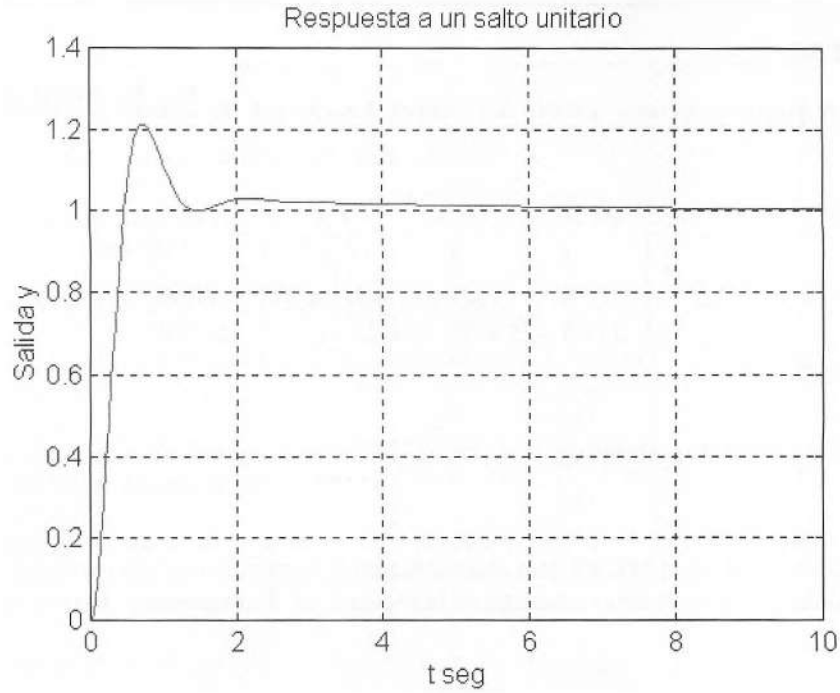


Figura 3-15

Un programa en MATLAB para representar las tres curvas x1, x2 y x3 respecto de t en un diagrama así como para hacerlo en gráficas separadas se da en el programa 3-12 en MATLAB. En la Figura 3-17 se muestra una gráfica de las tres curvas en un mismo diagrama. En las Figuras 3-18 y 3-19 se muestran gráficas separadas de x2 y x3 respectivamente.

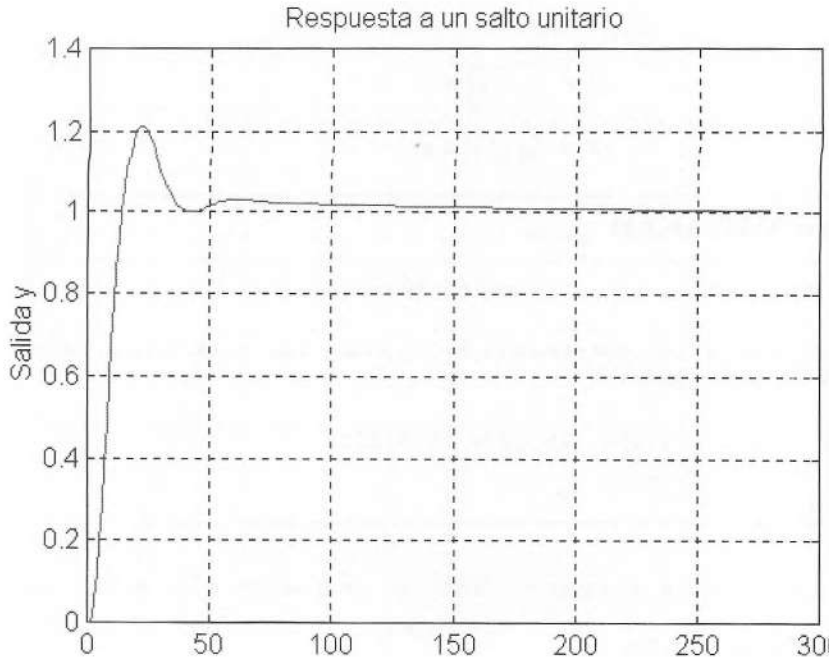


Figura 3-16

Programa en MATLAB 3-12

```
%-------Respuesta a un salto unitario-------

%*****Este programa es una continuación del programa 3-11.
%Vea el programa 3-11 para las matrices A,B,C y D y los vectores x e y *****

%*****Para representar x1 respecto de t, x2 respecto de t y x3 respecto de t
%en una misma gráfica, introduzca las siguientes órdenes *****

plot(t,x)
text(5,2.5,'x1')
text(5,-1.5,'x2')
text(5,-24.0,'x3')

%*****Añada una rejilla, título y etiquetas para los ejes x e y*****

grid
title('Curvas de respuesta a un salto unitario para x1, x2 y x3')
xlabel('t seg')
ylabel('x1, x2 y x3')
pause

%*****A continuación, representamos cada curva separadamente*****

%*****Como x1=y, la curva x1 respecto de t es la misma que la curva
%obtenida previamente de y resepecto de t *****

%*****Para representar x2 respecto de t, introduzca las siguientes órdenes*****

x2=[0 1 0]*x';plot(t,x2)
grid
title('Respuesta de x2 respecto de t')
xlabel('t seg')
ylabel('x2')
pause

%*****Para representar x3 respecto de t, introduzca las siguientes órdenes*****

x3=[0 0 1]*x';plot(t,x3)
grid
title('Respuesta de x3 respecto de t')
xlabel('t seg')
ylabel('x3')
```

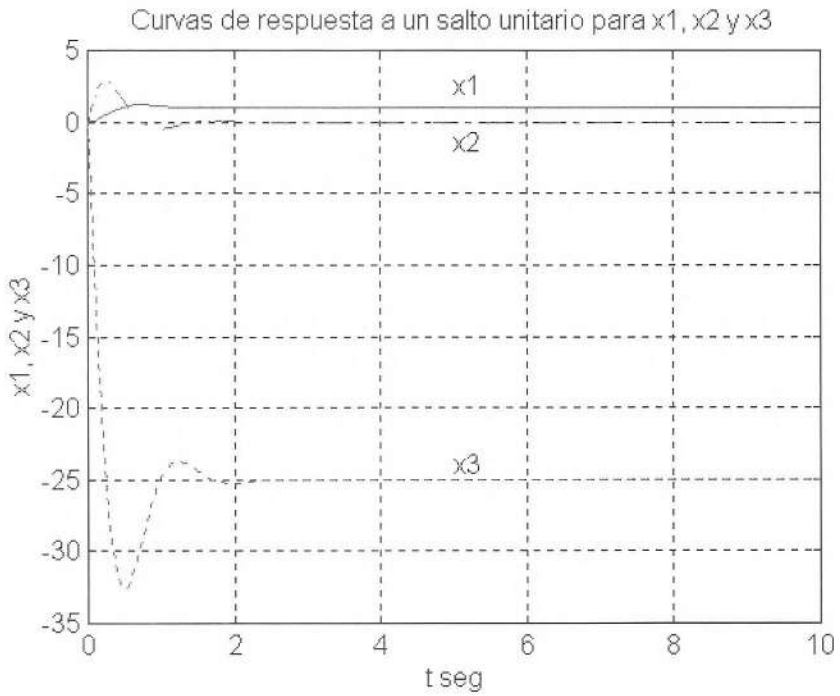


Figura 3-17

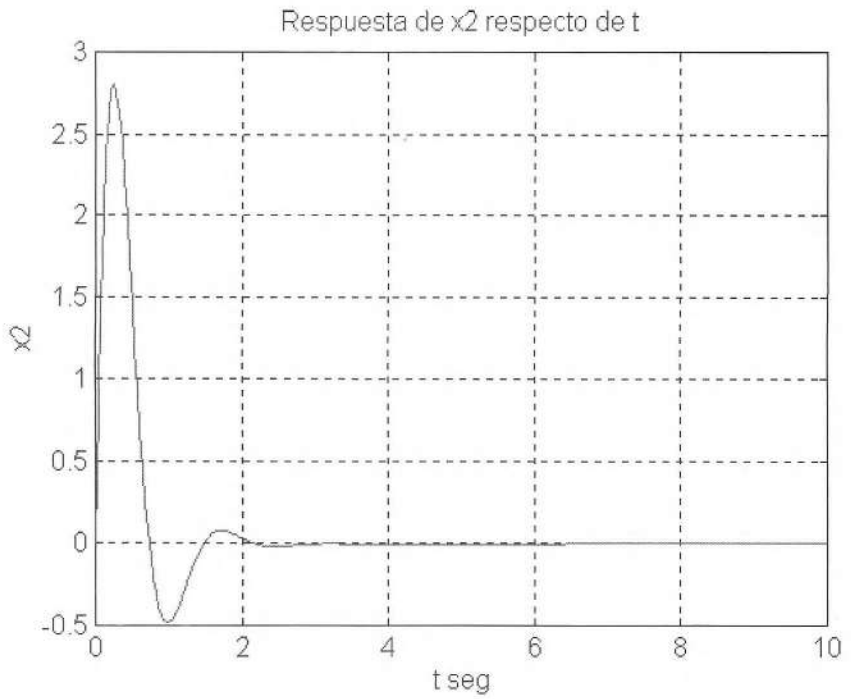


Figura 3-18

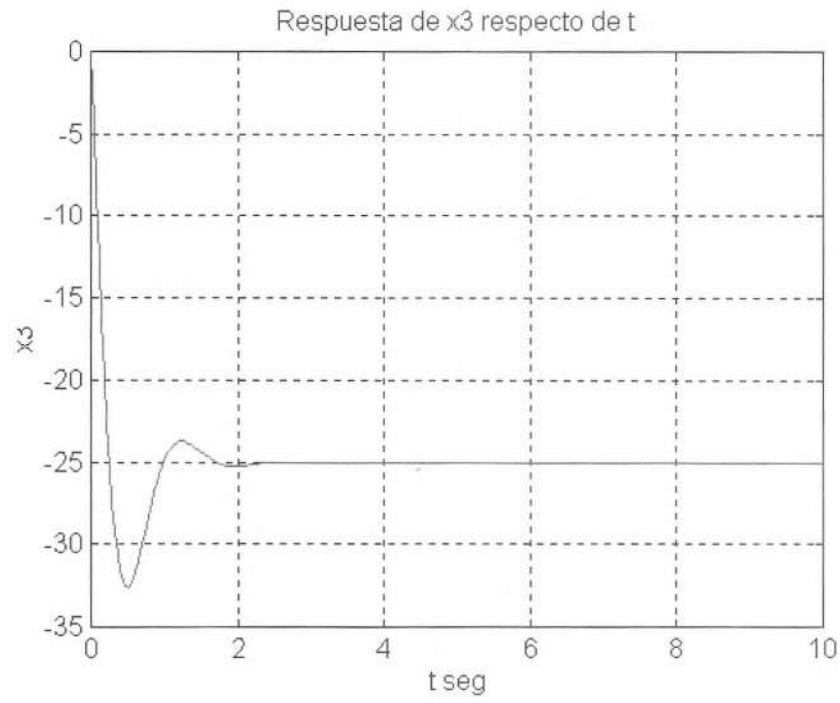


Figura 3-19

Respuesta a un salto (caso singular)

Como se ha dicho anteriormente, en la obtención de la respuesta a un salto unitario, algunos casos singulares originarán división por cero en los cálculos con MATLAB. Presentaremos otra vez un caso singular de este tipo.

Sea el sistema definido por las siguientes ecuaciones en el espacio de estado

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & -0.5 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} u$$

El programa 3-13 en MATLAB se puede utilizar para obtener la respuesta de $y(t)$ respecto de t cuando la entrada u es una función salto unitario. Observe que la orden

```
step (A, B, C, D)
```

no dará la respuesta a un salto. En este caso puede aparecer en la pantalla un mensaje de aviso 'Divide by zero'. Si esto sucede, para obtener la curva de respuesta en la pantalla, introduzca un conjunto específico de

instantes de tiempo de cálculo (tal como t = 0:0.1:12) e introduzca el tiempo t especificado por el usuario en la orden step tal como sigue:

step (A, B, C, D, 1, t)

Programa en MATLAB 3-13

```
%-------Respuesta a un salto unitario-------

%*****Obtendremos la respuesta a un salto unitario de un sistema
%definido en el espacio de estados*****

%*****Introduzca las matrices A,B,C y D*****

A=[-1 -0.5;1 0];
B=[0.5;0];
C=[1 0];
D=[0];

%*****Introduzca la orden de respuesta a un salto unitario*****

step(A,B,C,D);

%*****Si aparece este mensaje de aviso, entonces especifique los puntos
%de tiempo de cálculo (tal como t=0:0.1:12) e introduzca la siguiente
%orden de respuesta a un salto unitario: step(A,B,C,D,1,t)*****

t=0:0.1:12;
step(A,B,C,D,1,t);

%Añada la rejilla y el título de la gráfica

grid
title('Respuesta a un salto unitario de un sistema definido en el espacio de
estados')
```

En la Figura 3-20 se muestra la curva de respuesta resultante.

Observe que si introducimos la orden

step (A, B, C, D, t)

aparecerá en la pantalla un mensaje de aviso, 'Index into matrix is negative or zero'. Como este sistema tiene solamente una función de entrada u, introduzca el índice '1' en la orden step tal como step (A, B, C, D, 1, t).

Una forma alternativa para evitar división por cero es hacer un pequeño cambio en uno (o más) elementos de la matriz A. Por ejemplo, cambiando la matriz A a

$$\begin{bmatrix} -1 & -0.4999 \\ 1 & 0 \end{bmatrix}, \quad \begin{bmatrix} -1 & -0.5 \\ 0.9999 & 0 \end{bmatrix}$$

eliminaremos la dificultad

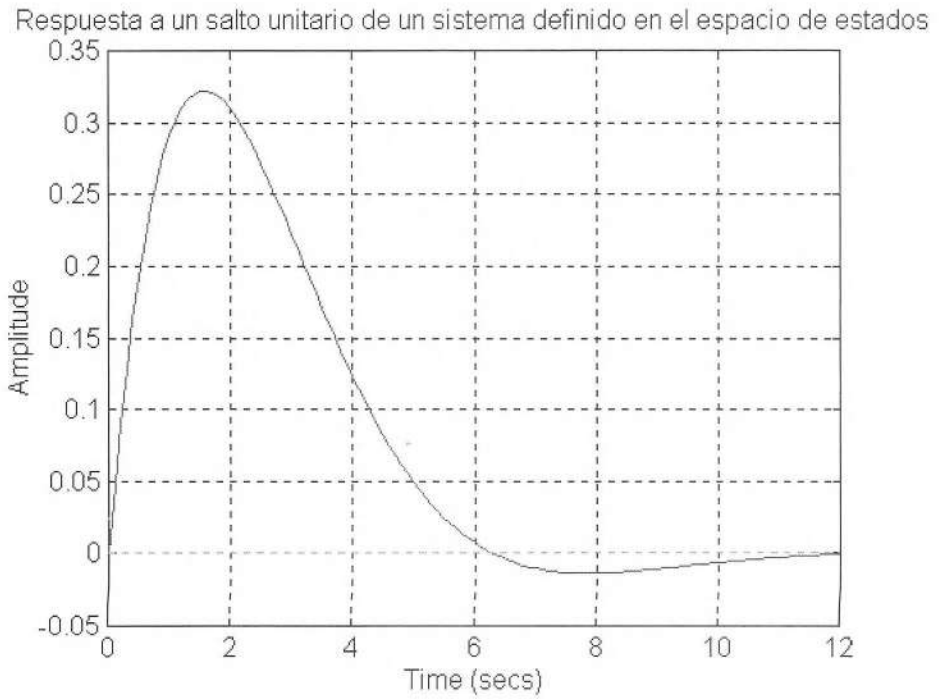


Figura 3-20

EJEMPLO 3-6

Considere el siguiente sistema

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ 6.5 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Obtenga las curvas de respuesta a un salto unitario.
En primer lugar calcularemos la expresión de la matriz de transferencia para el sistema:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

La matriz de transferencia *G(s)* es una matriz que relaciona *Y(s)* y U(s) de la forma siguiente:

$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s)$$

Tomando las transformadas de Laplace de las ecuaciones en el espacio de estado, obtenemos:

$$s\mathbf{X}(s) - \mathbf{x}(0) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s) \tag{3-2}$$
$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s) + \mathbf{D}\mathbf{U}(s) \tag{3-3}$$

Para derivar la matriz de transferencia, suponemos que $\mathbf{x}(0) = \mathbf{0}$. A partir de (3-2) obtenemos

$$\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{U}(s) \tag{3-4}$$

Sustituyendo (3-4) en (3-3), obtenemos

$$\mathbf{Y}(s) = [\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}]\mathbf{U}(s)$$

Así, la matriz de transferencia $G(s)$ viene dada por

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$

La matriz de transferencia $\mathbf{G}(s)$ para el sistema dado es:

$$\mathbf{G}(s) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} s+1 & 1 \\ -6.5 & s \end{bmatrix}^{-1}\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} = \frac{1}{s^2+s+6.5}\begin{bmatrix} s & -1 \\ 6.5 & s+1 \end{bmatrix} = \frac{1}{s^2+s+6.5}\begin{bmatrix} s-1 & s \\ s+7.5 & 6.5 \end{bmatrix}$$

Por tanto

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \dfrac{s-1}{s^2+s+6.5} & \dfrac{s}{s^2+s+6.5} \\ \dfrac{s+7.5}{s^2+s+6.5} & \dfrac{6.5}{s^2+s+6.5} \end{bmatrix}\begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix}$$

Como este sistema tiene dos entradas y dos salidas, se pueden definir cuatro funciones de transferencia dependiendo de que señales se consideran como entrada y salida. Observe que, cuando se considera la señal u_1 como entrada, suponemos que la señal u_2 es cero y viceversa. Las cuatro funciones de transferencia son

$$\frac{Y_1(s)}{U_1(s)} = \frac{s-1}{s^2+s+6.5}, \quad \frac{Y_2(s)}{U_1(s)} = \frac{s+7.5}{s^2+s+6.5}$$

$$\frac{Y_1(s)}{U_2(s)} = \frac{s}{s^2+s+6.5}, \quad \frac{Y_2(s)}{U_2(s)} = \frac{6.5}{s^2+s+6.5}$$

Las cuatro curvas individuales de respuesta a un salto se pueden representar utilizando la orden:

```
step(A, B, C, D)
```

el programa 3-14 en MATLAB genera tales curvas. En la Figura 3-21 se muestran las gráficas correspondientes

Programa en MATLAB 3-14

```
A=[-1 -1;6.5 0];
B=[1 1;1 0];
C=[1 0;0 1];
D=[0 0;0 0];
step(A,B,C,D);
```

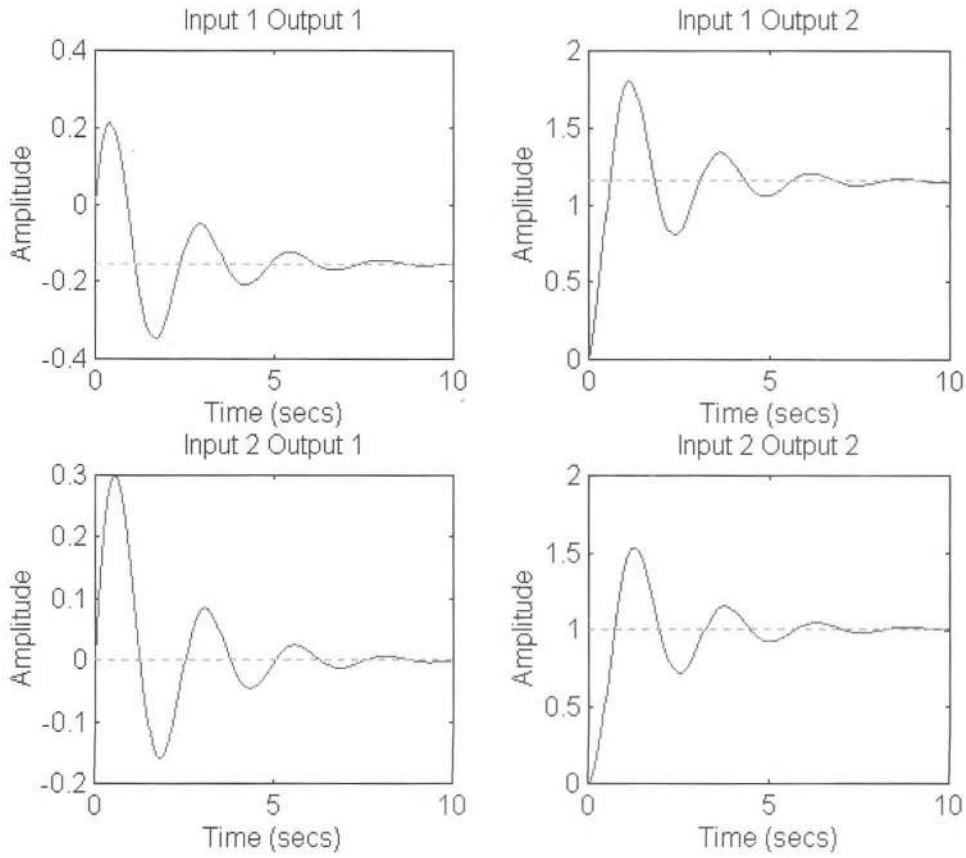


Figura 3-21

Para representar las dos curvas de respuesta a un salto de la entrada u_1 en un diagrama y las dos curvas de respuesta a un salto de la entrada u_2 en otro diagrama podemos utilizar las órdenes

```
step(A, B, C, D, 1)
```

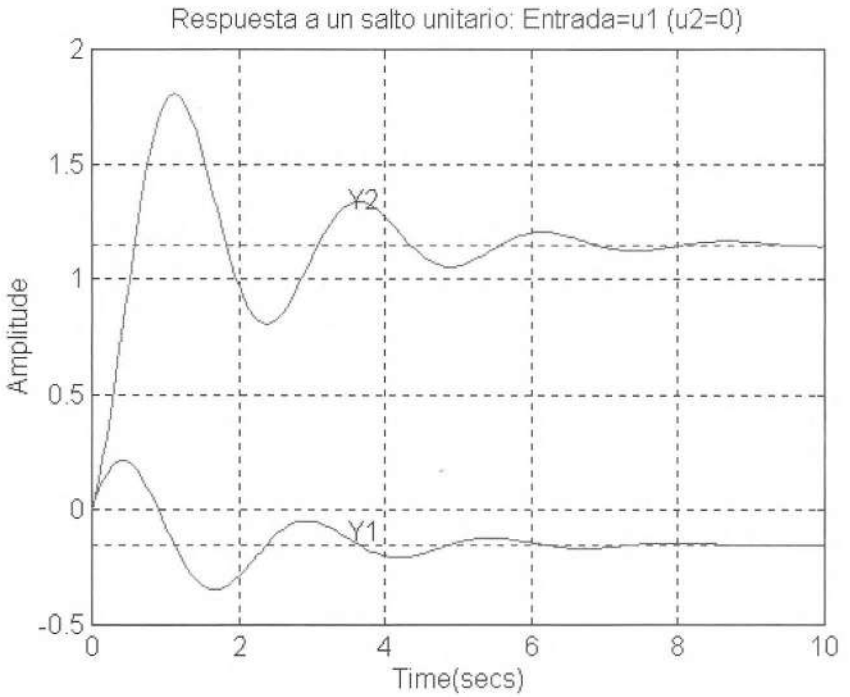
y

```
step(A, B, C, D, 2)
```

respectivamente. El programa 3-15 en MATLAB permite representar las dos curvas de respuesta a un salto de la entrada u_1 en un diagrama y las dos curvas de respuesta a un salto de la entrada u_2 en otro diagrama. La Figura 3-22 muestra los dos diagramas, conteniendo cada uno las dos curvas de respuesta a un salto.

Programa en MATLAB 3-15

```
%-------Respuesta a un salto unitario para un sistema definido en el espacio de
%estados-------

%*****En este programa se representan las curvas de respuesta a un salto
%unitario de un sistema que tiene dos entradas (u1 y u2) y dos salidas
%(y1 e y2)*****

%*****En primer lugar representaremos la curva de respuesta a un salto unitario
%cuando la entrada es u1. A continuación la curva de respuesta a un salto
%unitario cuando la entrada es u2*****

%*****Introduzca las matrices A,B,C y D*****


A=[-1 -1;6.5 0];
B=[1 1;1 0];
C=[1 0;0 1];
D=[0 0;0 0];

%*****Para representar las curvas de respuesta a un salto unitario cuando la
%entrada es u1, introduzca la orden 'step(A,B,C,D,1)'*****

step(A,B,C,D,1)
grid
title('Respuesta a un salto unitario: Entrada=u1 (u2=0)')
text(3.5,-0.1,'Y1')
text(3.5,1.35,'Y2')

%*****A continuación representamos las curvas de respuesta a un salto unitario
%cuando la entrada es u2. Introduzca la orden 'step(A,B,C,D,2)'*****

step(A,B,C,D,2)
grid
title('Respuesta a un salto unitario: Entrada=u2 (u1=0)')
text(3.5,0.06,'Y1')
text(3,1.1,'Y2')
```


Respuesta a un salto unitario: Entrada=u1 (u2=0)
Amplitude
Time(secs)
Y2
Y1
2
1.5
1
0.5
0
-0.5
0
2
4
6
8
10

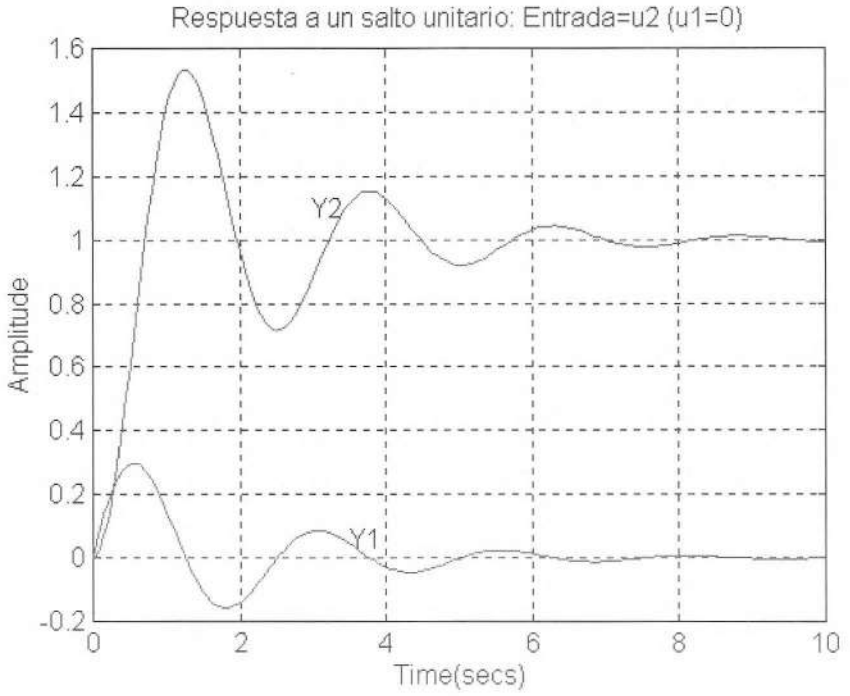

Respuesta a un salto unitario: Entrada=u2 (u1=0)
Amplitude
Time(secs)
Y2
Y1
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
2
4
6
8
10

Figura 3-22

3.3 RESPUESTA IMPULSIONAL

En esta sección se trata la respuesta impulsional de los sistemas de control. La idea básica ahora es que, cuando las condiciones iniciales son cero, la respuesta a un impulso unitario de $G(s)$ es la misma que la respuesta a un salto unitario de $sG(s)$.

Considere la respuesta a un impulso unitario del siguiente sistema:

$$\frac{C(s)}{R(s)} = G(s) = \frac{1}{s+1}$$

Como $R(s) = 1$ para la entrada impulso unitario, tenemos

$$\frac{C(s)}{R(s)} = G(s) = \frac{1}{s+1} = \left(\frac{s}{s+1}\right)\frac{1}{s}$$

Podemos así convertir la respuesta a un impulso unitario de $G(s)$ en la respuesta a un salto unitario de $sG(s)$. Si introducimos el numerador y denominador siguientes en el programa MATLAB

```
num = [1 0]
den = [1 1]
```

y utilizamos la orden de respuesta a un salto, tal como se hace en el programa 3-16 en MATLAB, podemos obtener una gráfica de la respuesta a un impulso unitario del sistema como se muestra en la Figura 3-23.

Programa en MATLAB 3-16

```
%-------Respuesta a un impulso unitario-------

%*****Para obtener la respuesta a un impulso unitario de un sistema de primer
%orden G(s)=1/(s+1), multiplicar s por G(s) y utilizar la orden de respuesta a
%un salto unitario*****

%*****Introduzca el numerador y el denominador de sG(s)*****

num=[1 0];
den=[1 1];

%*****Introduzca la orden de respuesta a un salto unitario*****

step(num,den)
grid
title('Respuesta a un impulso unitario de G(s)=1/(s+1)')
```

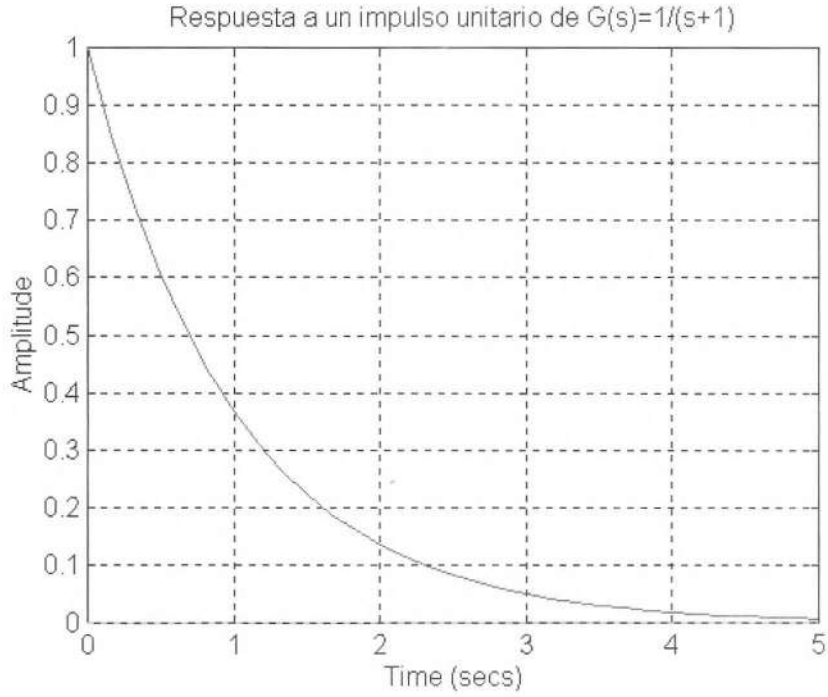


Figura 3-23

EJEMPLO 3-7

Sea la respuesta a un impulso unitario del sistema de segundo orden

$$\frac{C(s)}{R(s)} = G(s) = \frac{1}{s^2 + 0.2s + 1}$$

Para la entrada impulso unitario se tiene que $R(s) = 1$. Por tanto

$$\frac{C(s)}{R(s)} = \frac{1}{s^2 + 0.2s + 1} = \frac{s}{s^2 + 0.2s + 1}\frac{1}{s}$$

Considerando que la respuesta a un impulso unitario de $G(s)$ es la respuesta a un salto unitario de $sG(s)$, introduzca el numerador y denominador siguiente en el programa:

```
num = [0 1 0]
den = [1 0.2 1]
```

Un programa completo en MATLAB para obtener la respuesta a un impulso unitario de este sistema se da en el programa 3-17 en MATLAB. En la Figura 3-24 se muestra una gráfica de la respuesta a un impulso unitario del sistema.

Programa en MATLAB 3-17

```
%-------Respuesta a un impulso unitario-------

%*****Respuesta a un impulso unitario de G(s)=1/(s^2+0.2s+1)*****

%*****Para obtener la respuesta a un impulso unitario de G(s), multiplique
%s por G(s) y utilice la orden de respuesta a un salto unitario*****

%*****Introduzca el numerador y el denominador de sG(s)*****

num=[0 1 0];
den=[1 0.2 1];

%*****Introduzca la orden de respuesta a un salto unitario*****

t=0:0.1:50;
step(num,den,t)
grid
title('Respuesta a un impulso unitario de G(s)=1/(s^2+0.2s+1)')
```

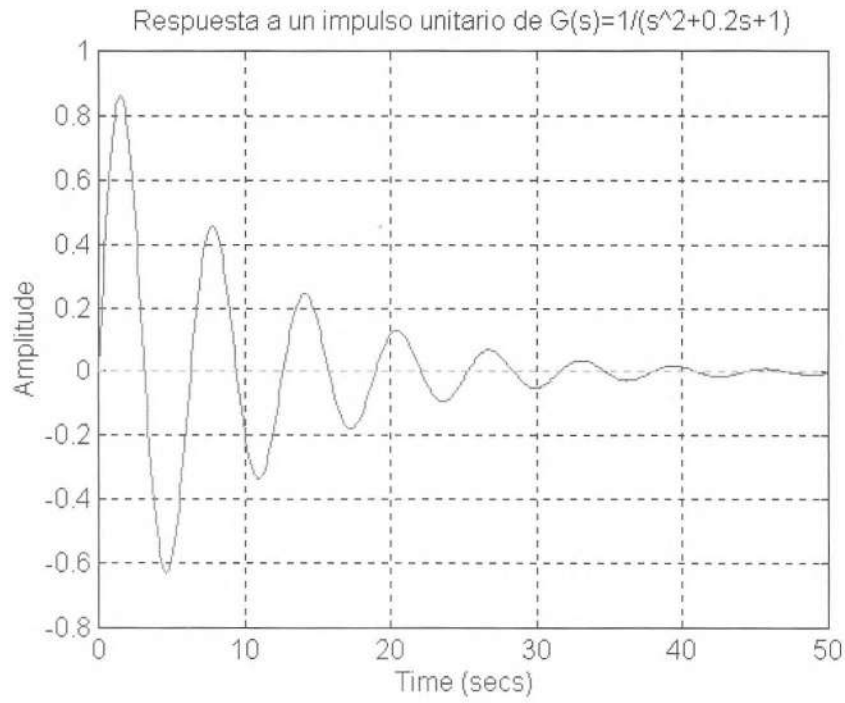


Figura 3-24

EJEMPLO 3-8

Sea la respuesta a un impulso unitario del sistema de segundo orden

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n^2 s + 1}$$

Para la entrada impulso unitario se tiene que $R(s) = 1$. Así

$$C(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n^2 s + 1} = \frac{\omega_n^2 s}{s^2 + 2\zeta\omega_n^2 s + 1}\frac{1}{s}$$

Considere el sistema normalizado donde $w_n = 1$. Entonces

$$C(s) = \frac{s}{s^2 + 2\zeta s + 1}\frac{1}{s}$$

Para obtener la respuesta a un impulso unitario, introduzca el siguiente numerador y denominador en el programa

```
num = [0 1 0]
den = [1 2*(zeta) 1]
```

Consideraremos cinco valores diferentes de zeta: $\zeta = 0.1, 0.3, 0.5, 0.7$ y 1.0.

Un programa en MATLAB para representar las cinco curvas de respuesta a un impulso unitario en un mismo diagrama se da en el programa 3-18 en MATLAB. En la Figura 3-25 se muestra el diagrama resultante.

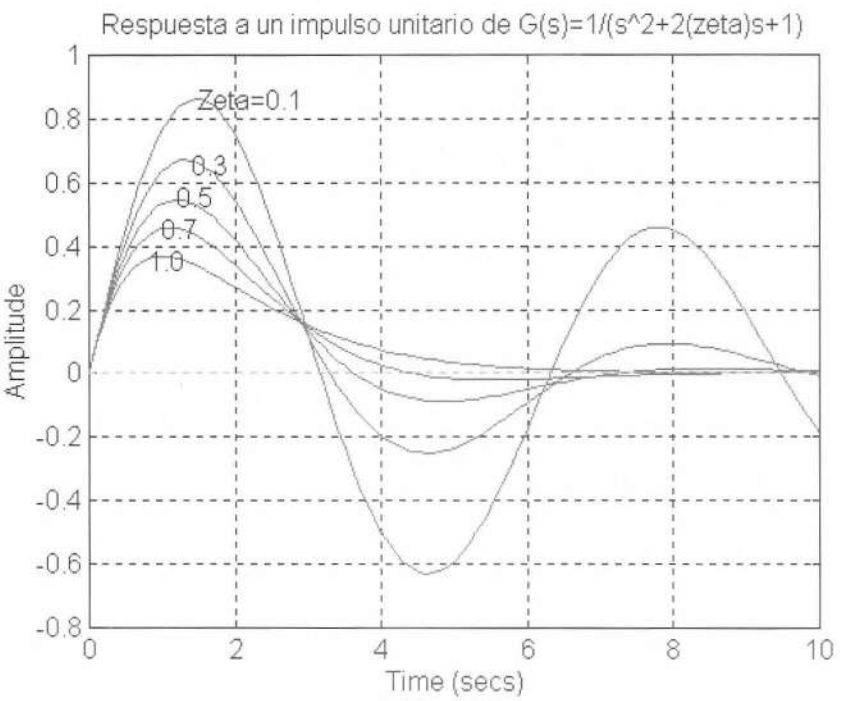


Figura 3-25

Programa en MATLAB 3-18

```
%-------Respuesta a un salto y a un impulso unitario-------

%*****Respuesta a un salto y a un impulso unitario de un sistema definido en
%el espacio de estados*****

%*****Introduzca las matrices A, B, C y D de las ecuaciones en el espacio de
%estados*****

A=[0        1        0        0
   0        0        1        0
   0        0        0        1
  -0.0073 -0.0878 -0.4791 -1.4791];
B=[0;0.0878;-0.0347;0.0166];
C=[1 0 0 0];
D=[0];

%*****Introduzca la siguiente orden de respuesta a un salto unitario*****

[y,x,t]=step(A,B,C,D);
plot(t,y)
grid
title('Respuesta a un salto unitario')
xlabel('t seg')
ylabel('salida y')

%*****La respuesta a un impulso unitario del sistema es la misma que la derivada
%de la respuesta a un salto unitario. Por esta razón, la respuesta a un impulso
%unitario de este sistema viene dada por y'=x2. Para representar la respuesta
%impulsional introduzca la orden siguiente*****

x2=[0 1 0 0]*x';plot(t,x2)
grid
title('Respuesta a un impulso unitario')
xlabel('t seg')
ylabel('Salida a un impulso unitario en la entrada, x2')
```

Si se desean utilizar marcas diferentes para las distintas curvas (tales como 'o', ':'), modifique el programa en MATLAB previo. La versión modificada se da como programa 3-19 en MATLAB. En la Figura 3-26 se muestran las gráficas resultantes de las curvas.

De las curvas de respuesta a un impulso unitario para diferentes valores de zeta, podemos concluir que si la respuesta impulsional c(t) no cambia de signo el sistema es amortiguado crítico o sobre amortiguado, en cuyo caso la correspondiente respuesta a un salto no presenta sobreelongación sino que aumenta o disminuye monotónicamente y tiende a un valor constante.

Programa en MATLAB 3-19

```
%-------Respuesta a un impulso unitario-------

%*****Respuesta a un impulso unitario del sistema normalizado de 2º orden
%G(s)=1/(s^2+0.2s+1)*****

%*****La respuesta a un impulso unitario se obtiene como la respuesta a un salto
%unitario de sG(s)*****

%*****Los valores que se consideran para el amortiguamiento (zeta) son:
%0.1, 0.3, 0.5, 0.7 y 1.0*****

%*****Introduzca el numerador y el denominador de sG(s) para zeta=0.1*****

num=[0 1 0];
den1=[1 0.2 1];

%*****Especifique los instantes de tiempo de cálculo (tales como t=0:0.1:10). A
%continuación introduzca la orden de respuesta a un salto unitario
%c=step(num,den,t)*****

t=0:0.1:10; c1=step(num,den1,t); plot(t,c1,'o')

%*****Mantenga esta gráfica y añada otra curva de respuesta a un impulso
%unitario*****

hold

%Current plot held

%*****Introduzca el denominador de sG(s) para zeta=0.3, 0.5, 0.7 y 1.0*****

den2=[1 0.6 1];den3=[1 1 1];den4=[1 1.4 1];den5=[1 2 1];

%*****Superponga en la gráfica que está activa las curvas de respuesta a un
%impulso unitario para zeta=0.3, 0.5, 0.7 y 1.0 introduciendo sucesivamente las
%órdenes de respuesta a un salto unitario step(num,den,t) y la orden de texto
%text(,,'')*****

c2=step(num,den2,t);plot(t,c2,'x')
c3=step(num,den3,t);plot(t,c3,'--')
c4=step(num,den4,t);plot(t,c4,'-')
c5=step(num,den5,t);plot(t,c5,':')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid, title('Respuesta a un impulso unitario de G(s)=1/(s^2+2(zeta)s+1)')
xlabel('t seg'), ylabel('Salidas c1, c2, c3, c4, c5')

%*****Borrar el mantener las gráficas*****

hold

%Current plot released
```

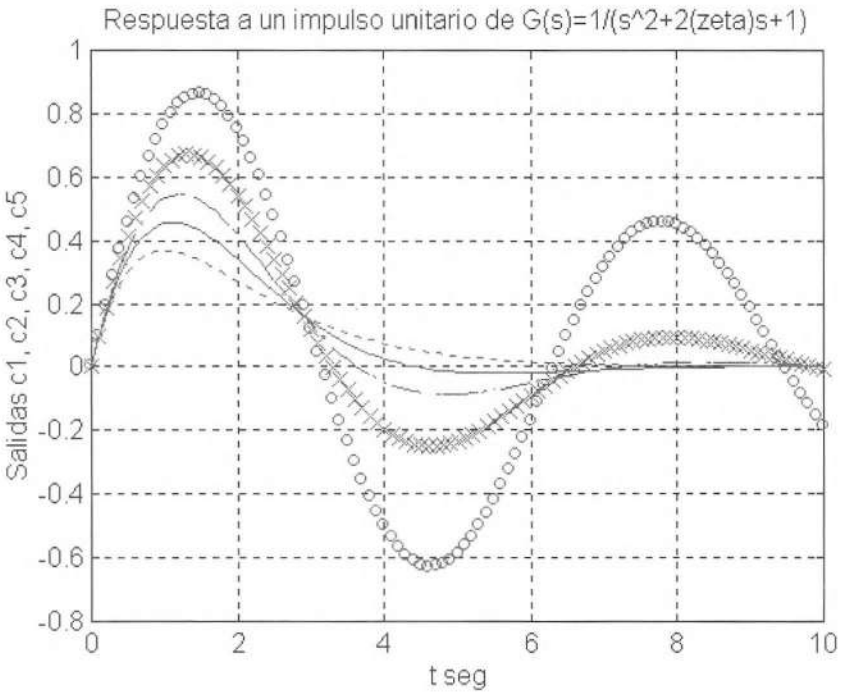


Figura 3-26

EJEMPLO 3-9

Obtener la respuesta a un salto y a un impulso unitarios del siguiente sistema:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -0.0073 & -0.0878 & -0.4791 & -1.4791 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.0878 \\ -0.0347 \\ 0.0166 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Para obtener la respuesta a un salto unitario de este sistema, utilice la orden

$$[y, x, t] = \text{step}(A, B, C, D)$$

Como la respuesta a un impulso unitario es la derivada de la respuesta a un salto unitario, la derivada de la salida (y = x1) dará la respuesta a un impulso unitario. De la ecuación de estado, vemos que la derivada de y es x2. Por lo tanto x2 respecto de t dará la respuesta a un impulso unitario. x2 puede obtenerse de

$$x2 = \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix} \times x'$$

El programa 3-20 en MATLAB dará ambas respuestas a un salto y a un impulso unitario.

La Figura 3-27 muestra la gráfica de la respuesta a un salto unitario y la Figura 3-28 a un impluso unitario.

Programa en MATLAB 3-20

```
%-------Respuesta a un salto y a un impulso unitario-------

%*****Respuesta a un salto y a un impulso unitario de un sistema definido en
%el espacio de estados*****

%*****Introduzca las matrices A, B, C y D de las ecuaciones en el espacio de
%estados*****

A=[0        1        0        0
   0        0        1        0
   0        0        0        1
  -0.0073 -0.0878 -0.4791 -1.4791];
B=[0;0.0878;-0.0347;0.0166];
C=[1 0 0 0];
D=[0];

%*****Introduzca la siguiente orden de respuesta a un salto unitario*****

[y,x,t]=step(A,B,C,D);
plot(t,y)
grid
title('Respuesta a un salto unitario')
xlabel('t seg')
ylabel('salida y')

%*****La respuesta a un impulso unitario del sistema es la misma que la derivada
%de la respuesta a un salto unitario. Por esta razón, la respuesta a un impulso
%unitario de este sistema viene dada por y'=x2. Para representar la respuesta
%impulsional introduzca la orden siguiente*****

x2=[0 1 0 0]*x';plot(t,x2)
grid
title('Respuesta a un impulso unitario')
xlabel('t seg')
ylabel('Salida a un impulso unitario en la entrada, x2')
```

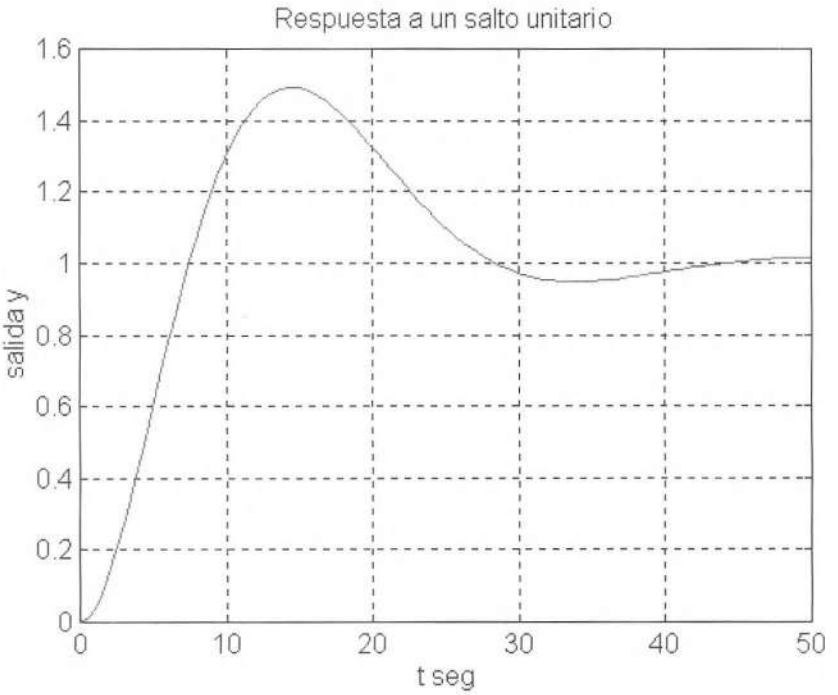


Figura 3-27

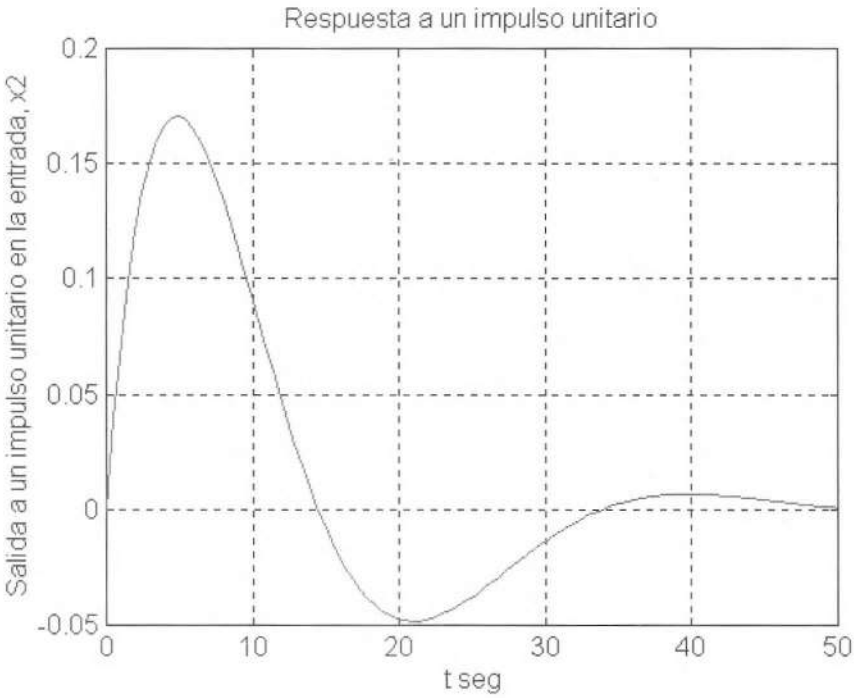


Figura 3-28

3.4 RESPUESTA A UNA ENTRADA EN RAMPA

Para obtener la respuesta a una entrada en rampa de la función de transferencia del sistema $G(s)$, divida $G(s)$ por s y utilice la orden de respuesta a un salto. Por ejemplo, sea el sistema en lazo cerrado

$$\frac{C(s)}{R(s)} = \frac{1}{s^2 + s + 1}$$

Para una entrada en rampa unitaria se tiene que $R(s) = 1/s^2$. Por tanto

$$C(s) = \frac{1}{s^2 + s + 1}\frac{1}{s^2} = \frac{1}{(s^2 + s + 1)s}\frac{1}{s} \quad \text{OJO}$$

Para obtener la respuesta a una entrada en rampa unitaria, introduzca el siguiente numerador y denominador en el programa en MATLAB,

```
num = [0 0 0 1];
den = [1 1 1 0];
```

y utilice la orden de respuesta a un salto. Véase el programa 3-21 en MATLAB. En la Figura 3-29 se muestra la gráfica obtenida utilizando este programa.

Programa en MATLAB 3-21

```
%-------Respuesta a una entrada unitaria en rampa-------

%*****La respuesta a una entrada unitaria en rampa se obtiene como la respuesta
%a un salto unitario de G(s)/s*****

%*****Introduzca el numerador y el denominador de G(s)/s *****

num=[0 0 0 1];
den=[1 1 1 0];

%*****Especifique los instantes de tiempo de cálculo (tales como t=0:0.1:7).
%A continuación introduzca la orden de respuesta a un salto unitario
%c=step(num,den,t)*****

t=0:0.1:7;
c=step(num,den,t);

%*****Al representar la respuesta a una rampa, añada a la gráfica la entrada de
%referencia. La entrada de referencia es t. Incluya como argumentos de la orden
%plot lo siguiente: t,t,'-'. La orden plot en este caso es como sigue:
%plot(t,c,'o',t,t,'-')*****

plot(t,c,'o',t,t,'-')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid
title('Respuesta a una rampa unitaria del sistema G(s)=1/(s^2+s+1)')
xlabel('t seg')
ylabel('Salida c')
```

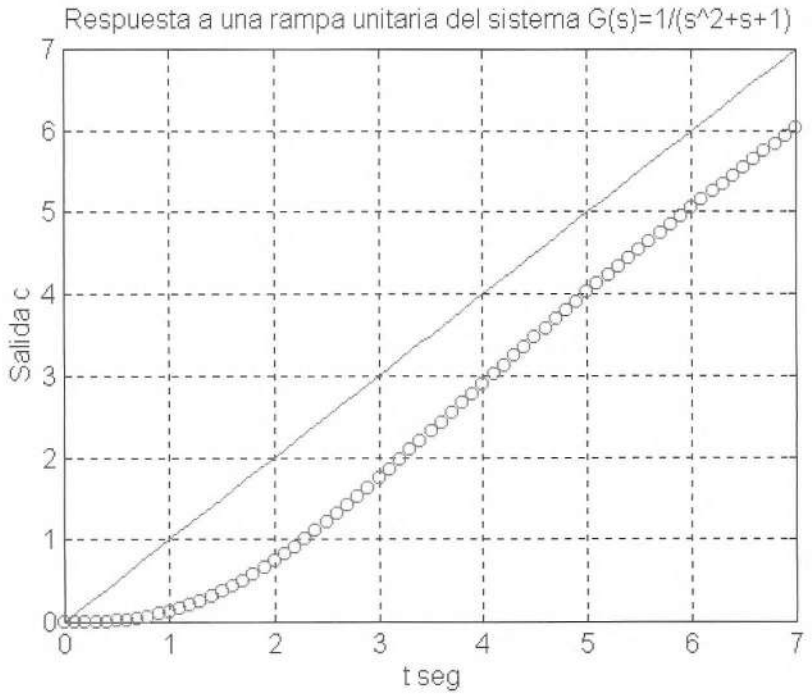


Figura 3-29

A continuación, trataremos la respuesta a una entrada en rampa unitaria del sistema representado en el espacio de estados. Sea el sistema descrito por

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

donde

$$A = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = [0], \quad x(0) = 0$$

Cuando las condiciones iniciales son cero, la respuesta a una entrada en rampa unitaria es la integral de la respuesta a un salto unitario. Por tanto, la respuesta a una rampa se puede expresar mediante

$$z = \int_0^t y\,dt \tag{3-5}$$

De (3-5), obtenemos

$$\dot{z} = y = x_1 \tag{3-6}$$

Definamos

$$z = x_3$$

Entonces (3-6) se transforma en

$$\dot{x}_3 = x_1 \tag{3-7}$$

Combinando (3-7) con la ecuación original en el espacio de estados, obtenemos

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} u$$

$$z = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

que puede escribirse como

$$\dot{\mathbf{x}} = \mathbf{AA}\mathbf{x} + \mathbf{BB}u$$
$$z = \mathbf{CC}\mathbf{x} + DDu$$

donde

$$\mathbf{AA} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \left[\begin{array}{cc} \mathbf{A} & \begin{matrix} 0 \\ 0 \end{matrix} \\ \mathbf{C} & 0 \end{array}\right], \quad \mathbf{BB} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \left[\begin{array}{c} \mathbf{B} \\ \hline 0 \end{array}\right], \quad \mathbf{CC} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, \quad DD = [0]$$

Observe que x_3 es el tercer elemento de x. Una gráfica de la curva de respuesta $z(t)$ a una entrada rampa unitaria se puede obtener introduciendo en el computador el programa 3-22 de MATLAB. En la Figura 3-30 se muestra una gráfica de la respuesta a una entrada en rampa unitaria obtenida de dicho programa.

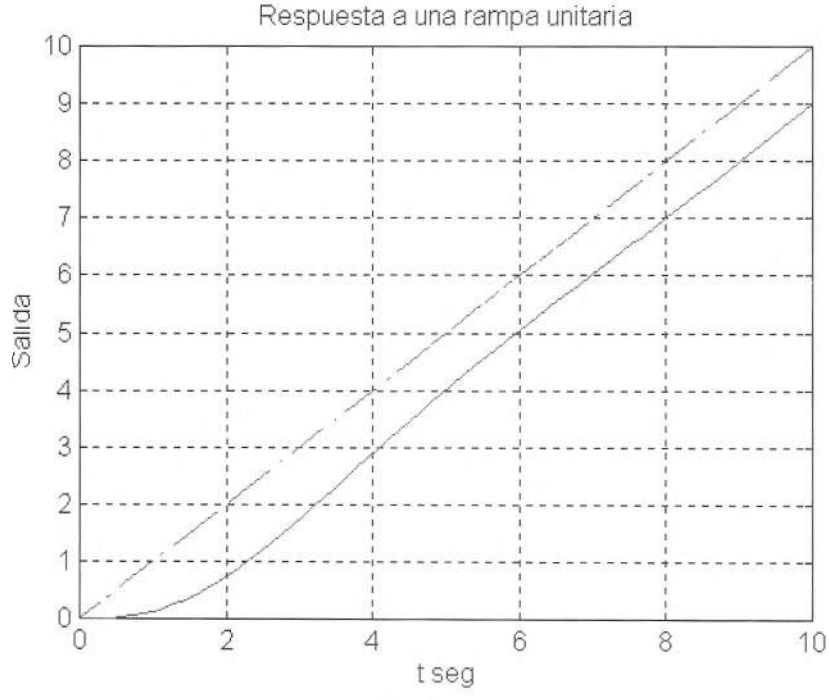


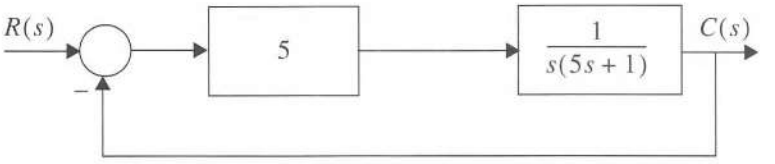
Figura 3-30

Programa en MATLAB 3-22

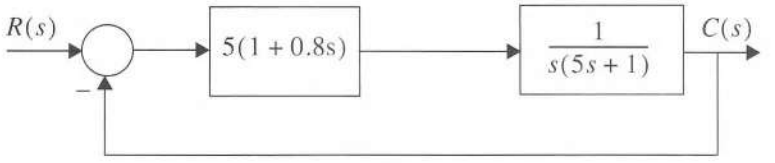
```
%-------Respuesta a una entrada unitaria en rampa-------

%*****La respuesta a una entrada unitaria en rampa se obtiene añadiendo una
%nueva variable de estado x3. La dimensión de la ecuación de estado se amplia en
%uno*****

%*****Introduzca las matrices A,B,C y D de la ecuación de estado original y de
%la ecuación de salida*****

A=[0 1;-1 -1];
B=[0;1];
C=[1 0];
D=[0];


%*****Introduzca las matrices AA,BB,CC y DD de las nuevas ecuaciones de estado
%y de salida ampliadas*****

AA=[A zeros(2,1);C 0];
BB=[B;0];
CC=[0 0 1];
DD=[0];

%*****Introduzca la orden de respuesta a un salto unitario;
%[z,x,t]=step(AA,BB,CC,DD)*****

[z,x,t]=step(AA,BB,CC,DD);

%*****Para representar x3 añada la entrada de una rampa unitaria t en la
%gráfica, introduciendo la orden siguiente: plot(t,x3,t,t,'--')*****

x3=[0 0 1]*x';plot(t,x3,t,t,'--')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid
title('Respuesta a una rampa unitaria')
xlabel('t seg')
ylabel('Salida')
```

EJEMPLO 3-10

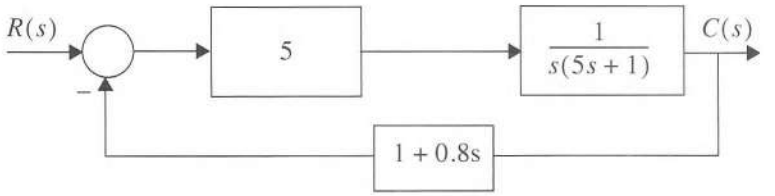
La Figura 3-31 muestra tres sistemas. El sistema I es un servo posicional. El sistema II es un servo posicional que utiliza una acción de control proporcional-derivativa. El sistema III es un servo posicional que utiliza una realimentación de velocidad o realimentación tacométrica. Comparemos las respuestas a un salto, un impulso y una rampa unitaria de los tres sistemas.



Sistema I



Sistema II



Sistema III

Figura 3-31

La función de transferencia en lazo cerrado para el sistema I es

$$\frac{C_1(s)}{R(s)} = \frac{1}{s^2 + 0.2s + 1}$$

La función de transferencia en lazo cerrado para el sistema II es

$$\frac{C_{11}(s)}{R(s)} = \frac{0.8s + 1}{s^2 + s + 1}$$

La función de transferencia en lazo cerrado para el sistema III es

$$\frac{C_{111}(s)}{R(s)} = \frac{1}{s^2 + s + 1}$$

Compararemos en primer lugar las respuestas a un salto unitario de los sistemas I, II y III. Las respuestas a un salto unitario se pueden obtener introduciendo el programa 3-23 en MATLAB. En la Figura 3-32 se muestran las curvas de respuesta a un salto unitario resultantes. Obsérvese de esta gráfica que el sistema que utiliza la acción de control proporcional-derivativa exhibe el tiempo de subida más corto. El sistema con realimentación de velocidad tiene la menor sobreelongación o la mejor estabilidad relativa de los tres sistemas.

Si se desea representar las tres curvas de respuesta a un salto unitario con líneas continuas, introduzca el programa 3-24 en MATLAB. En la Figura 3-33 se muestran las gráficas resultantes.

A continuación, obtendremos las curvas de respuesta a un impulso unitario de estos tres sistemas. Las curvas de respuesta a un impulso unitario se pueden representar utilizando el programa 3-25 en MATLAB. En la Figura 3-34 se muestran las gráficas resultantes.

Programa en MATLAB 3-23

```
%-------Respuesta a un salto-

%*****Introduzca el numerador y el denominador de los Sistemas I, II y III*****

num1=[0 0 1];
den1=[1 0.2 1];
num2=[0 0.8 1];
den2=[1 1 1];
num3=[0 0 1];
den3=[1 1 1];

%*****Especifique los instantes de tiempo de cálculo (tales como t=0:0.1:10).
%A continuación introduzca la orden de respuesta a un salto
%c=step(num,den,t)*****

t=0:0.1:10;
c1=step(num1,den1,t);
plot(t,c1,'o')

%*****Mantenga esta gráfica y añada otra curva de respuesta a un salto
%unitario*****

hold

%Current plot held

c2=step(num2,den2,t);
plot(t,c2,'x')
c3=step(num3,den3,t);
plot(t,c3,'--')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid
title('Respuesta a un salto unitario para los Sistemas I, II y III')
xlabel('t seg')
ylabel('salidas c1, c2 y c3')

%*****Borrar el mantener las gráficas*****

hold

%Current plot released
```

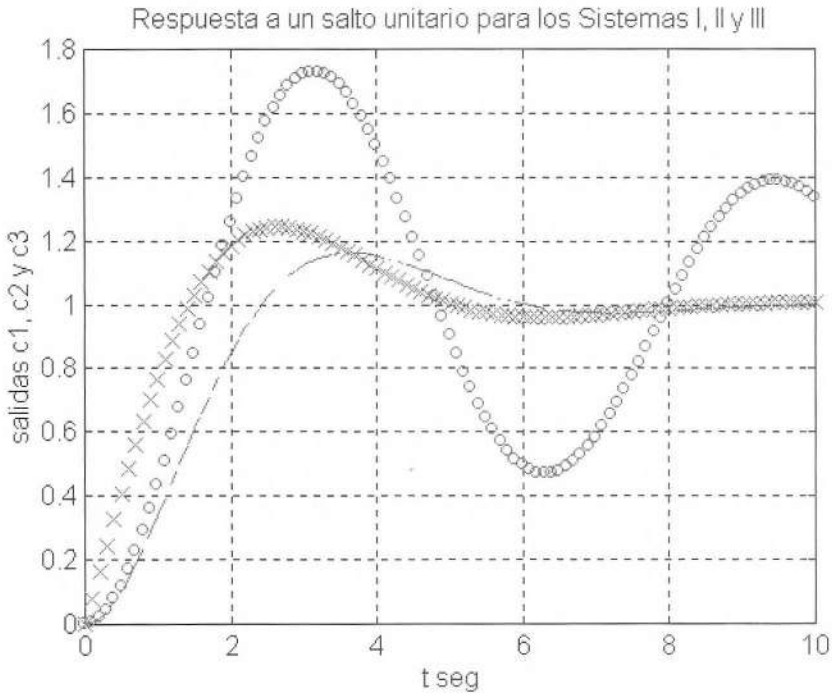


Figura 3-32

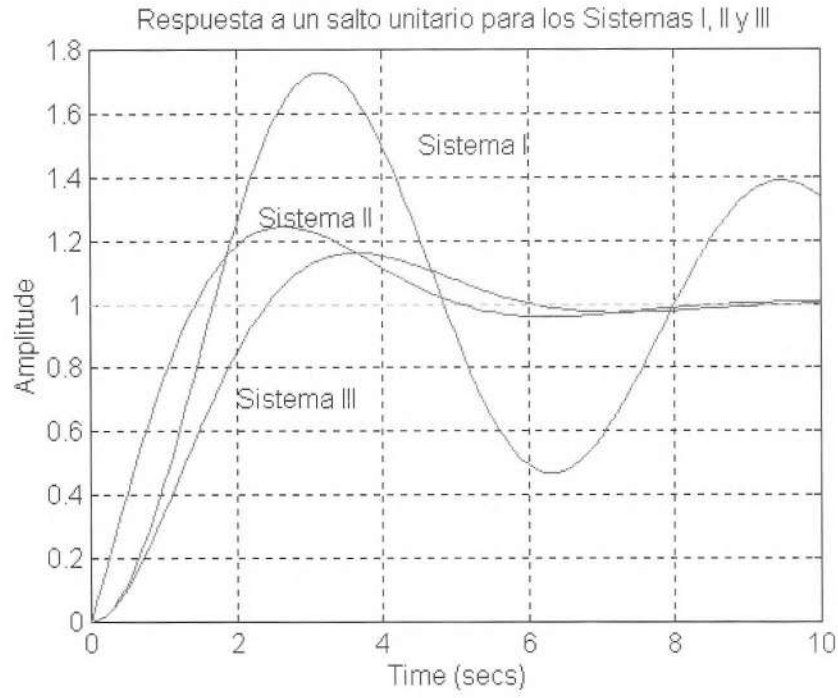


Figura 3-33

Programa en MATLAB 3-24

```
%-------Respuesta a un salto-------

%*****Introduzca el numerador y el denominador de los tres sistemas*****

num1=[0 0 1];
den1=[1 0.2 1];
num2=[0 0.8 1];
den2=[1 1 1];
num3=[0 0 1];
den3=[1 1 1];

%*****Especifique los instantes de tiempo de cálculo (tales como t=0:0.1:10).
%A continuación introduzca la orden de respuesta a un salto step(num,den,t)
%y la orden de texto text(,,'')*****

t=0:0.1:10;
step(num1,den1,t);
text(4.5,1.5,'Sistema I')

%*****Mantenga esta gráfica y añada otra curva de respuesta a un salto
%unitario*****

hold

%Current plot held

step(num2,den2,t);
text(2.3,1.27,'Sistema II')
step(num3,den3,t);
text(2,0.7,'Sistema III')

%*****Introduzca la rejilla y el título de la gráfica*****

grid
title('Respuesta a un salto unitario para los Sistemas I, II y III')

%*****Borrar el mantener las gráficas*****

hold

%Current plot released
```

Programa en MATLAB 3-25

```
%-------Respuesta a un impulso unitario-------

%******La respuesta a un impulso unitario se obtiene como la respuesta a un salto
%unitario de sG(s)*****

%******Introduzca el numerador y el denominador de sG1(s), sG2(s) y sG3(s)*****

num1=[0 1 0]; den1=[1 0.2 1];
num2=[0.8 1 0]; den2=[1 1 1];
num3=[0 1 0]; den3=[1 1 1];

%******Especifique los instantes de tiempo de cálculo (tales como t=0:0.1:10).
%A continuación introduzca la orden de respuesta a un salto c=step(num,den,t)
%y la orden plot*****

t=0:0.1:10;
c1=step(num1,den1,t);
c2=step(num2,den2,t);
c3=step(num3,den3,t);
plot(t,c1,'o',t,c2,'x',t,c3,'--')

%******Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid, title('Respuesta a un impulso unitario para los Sistemas I, II y III')
xlabel('t seg'), ylabel('salidas c1, c2 y c3')
```

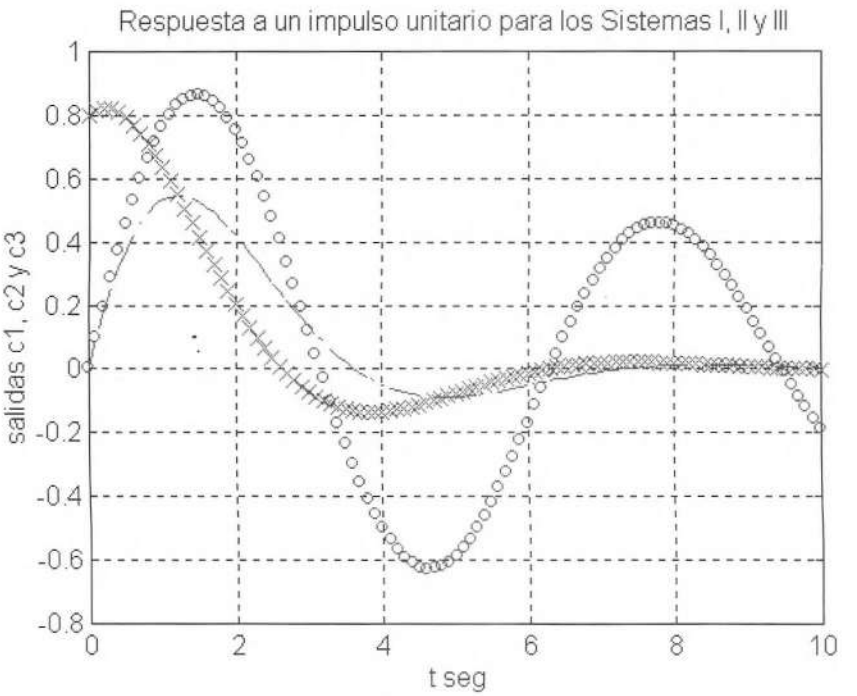


Figura 3-34

Finalmente, para obtener una gráfica de las curvas de respuesta a una entrada en rampa unitaria, introduzca en el computador el programa 3-26 en MATLAB. En la Figura 3-35 se muestra la gráfica resultante de estas tres curvas de respuesta a una entrada en rampa unitaria. De esta gráfica, vemos que el sistema II tiene la ventaja de una respuesta más rápida y un error menor en estado estacionario para una entrada en rampa.

La razón principal de porqué el sistema que utiliza la acción de control proporcional-derivativa tiene una característica de respuesta superior es que el control derivativo responde a la velocidad de cambio de la señal de error y puede producir una acción correctiva antes que la magnitud del error se haga grande.

Observe que la salida del sistema III es la salida del sistema II retardada por el filtro de primer orden siguiente $1/(1 + 0.8s)$. La Figura 3-36 muestra la relación entre las salidas del sistema II y del sistema III.

Programa en MATLAB 3-26

```
%-------Respuesta a una rampa unitaria-------

%*****La respuesta a un impulso unitario se obtiene como la respuesta
%a un salto unitario de G(s)/s*****

%*****Introduzca el numerador y el denominador de G1(s)/s, G2(s)/s y
%G3(s)/s*****

num1=[0 0 0 1];
den1=[1 0.2 1 0];
num2=[0 0 0.8 1];
den2=[1 1 1 0];
num3=[0 0 0 1];
den3=[1 1 1 0];

%*****Especifique los instantes de tiempo de cálculo (tales como t=0:0.1:7).
%A continuación introduzca la orden de respuesta a un salto c=step(num,den,t).
%Añada la función de entrada en rampa en la gráfica introduciendo la orden
%plot(t,t,'-')*****

t=0:0.1:7;
c1=step(num1,den1,t);
c2=step(num2,den2,t);
c3=step(num3,den3,t);
plot(t,c1,'o',t,c2,'x',t,c3,'--',t,t,'-')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid
title('Respuesta a una entrada en rampa unitaria para los Sistemas I, II y III')
xlabel('t seg')
ylabel('salidas c1, c2 y c3')
```

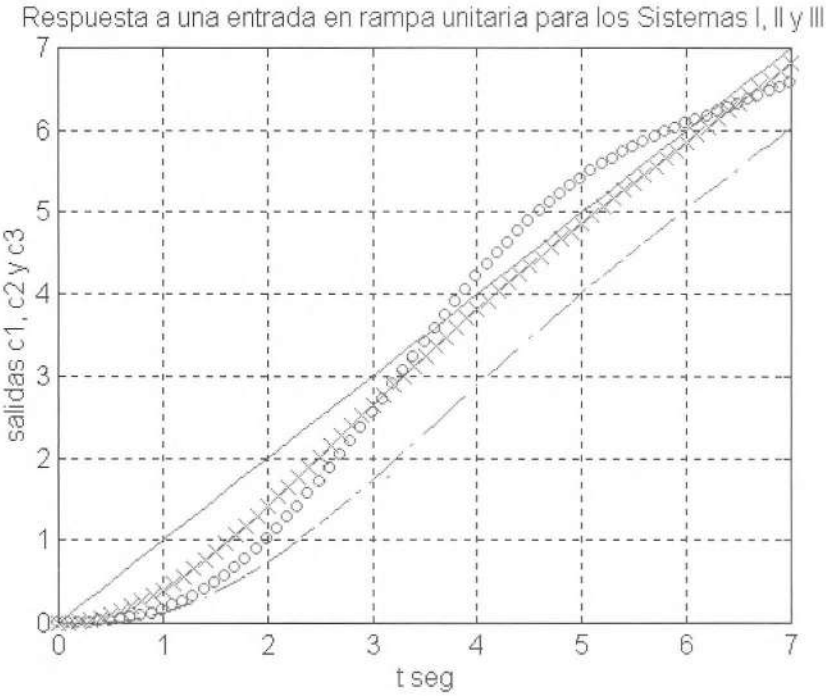


Figura 3-35

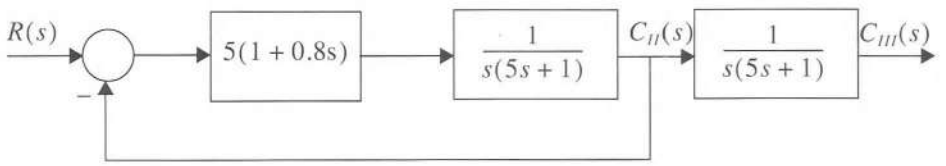


Figura 3-36

3.5 TRANSFORMACIÓN DE MODELOS DE SISTEMAS

En esta sección consideraremos la transformación del modelo del sistema del formato función de transferencia a espacio de estados y viceversa. Comenzaremos con la transformación de función de transferencia a espacio de estados.

Escribamos la función de transferencia en lazo cerrado como

$$\frac{Y(s)}{U(s)} = \frac{\text{polinomio del numerador en s}}{\text{polinomio del denominador en s}} = \frac{num}{den}$$

Una vez que tenemos esta expresión de la función de transferencia, la siguiente orden en MATLAB

[A, B, C, D] = tf2ss(num, den)

dará una representación en el espacio de estados. Es importante observar que la representación en el espacio de estados para cualquier sistema no es única. Hay muchas (infinitas) representaciones en el espacio de estados para el mismo sistema. La orden de MATLAB da una de estas posibles representaciones.

Formulación en el espacio de estados de sistemas dados como función de transferencia

Sea la función de transferencia del sistema

$$\frac{Y(s)}{U(s)} = \frac{s}{(s+10)(s^2+4s+16)} = \frac{s}{s^3+14s^2+56s+160} \tag{3-8}$$

Hay muchas (infinitas) posibles representaciones en el espacio de estados para este sistema. Una representación posible es

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -160 & -56 & -14 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -14 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u$$

Otra representación posible (entre las infinitas alternativas) es

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -14 & -56 & -160 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u \tag{3-9}$$

$$y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u \tag{3-10}$$

MATLAB transforma la función de transferencia dada por (3-8) en la representación en el espacio de estados dada por (3-9) y (3-10). Para el ejemplo del sistema considerado aquí, el programa 3-27 en MATLAB producirá las matrices A, B, C y D.

Programa en MATLAB 3-27

```
num=[0 0 1 0];
den=[1 14 56 160];
[A,B,C,D]=tf2ss(num,den)

A =

   -14   -56  -160
     1     0     0
     0     1     0


B =

     1
     0
     0


C =

     0     1     0


D =

     0
```

EJEMPLO 3-11

Sea el sistema

$$\frac{Y(s)}{U(s)} = \frac{25.04s + 5.008}{s^3 + 5.03247s^2 + 25.1026s + 5.008}$$

Esta función de transferencia se puede transformar en una representación en el espacio de estados introduciendo el programa 3-28 en MATLAB. Las ecuaciones en el espacio de estados para el sistema obtenido mediante el uso de este programa son

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -5.0325 & -25.1026 & -5.008 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 25.04 & 5.008 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u$$

Programa en MATLAB 3-28

```
%-------Transformación del formato función de transferencia al espacio de
%estados-------

%*****La función de transferencia se puede transformar al formato del espacio
%de estados introduciendo la siguiente orden de transformación:
%[A,B,C,D]=tf2ss(num,den)*****

%*****Introduzca el numerador y el denominador de la función de
%transferencia*****


num=[0 0 25.04 5.008];
den=[1 5.03247 25.1026 5.008];

%*****Introduzca la orden de transformación*****

[A,B,C,D]=tf2ss(num,den)

A =

   -5.0325  -25.1026   -5.0080
    1.0000         0         0
         0    1.0000         0


B =

     1
     0
     0


C =

         0   25.0400    5.0080


D =

     0
```

Transformación del espacio de estados a función de transferencia

Para obtener la función de transferencia de las ecuaciones en el espacio de estados, utilice la siguiente orden:

[num, den] = ss2tf(A, B, C, D, iu)

iu debe especificarse para sistemas que tengan más de una entrada. Por ejemplo, si el sistema tiene tres entradas (u1, u2, u3), entonces iu debe ser 1, 2 o 3, donde 1 implica u1, 2 implica u2 y 3 implica u3.

Si el sistema tiene solamente una entrada, entonces

```
[num, den] = ss2tf 8(A, B, C, D)
```

o

```
[num, den] = ss2tf 8(A, B, C, D, 1)
```

se pueden utilizar. (Véase el Ejemplo 3-12 y el programa 3-29 en MATLAB).

Para el caso en que el sistema tiene múltiples entradas y salidas, véase el Ejemplo 3-13.

EJEMPLO 3-12

Obtener la función de transferencia del sistema definido por las siguientes ecuaciones en el espacio de estados:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -5.008 & -25.1026 & -5.03247 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 25.04 \\ -121.005 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

El programa 3-29 en MATLAB producirá la función de transferencia para el sistema dado. La función de transferencia obtenida viene dada por:

$$\frac{Y(s)}{U(s)} = \frac{25.04s + 5.008}{s^3 + 5.0325s^2 + 25.1026s + 5.008}$$

Programa en MATLAB 3-29

```
%-------Transformación del formato espacio de estados al de función de
%transferencia-------

%*****La transformación de las ecuaciones del espacio de estados en función de
%transferencia se efectua introduciendo la orden de transformación:
%[num,den]=ss2tf(A,B,C,D)*****

%*****Introduzca las matrices A,B,C y D de las ecuaciones en el espacio de
%estados*****


A=[0 1 0;0 0 1;-5.008 -25.1026 -5.03247];
B=[0; 25.04; -121.005];
C=[1 0 0];
D=[0];

%*****Introduzca la siguiente orden de transformación*****

[num,den]=ss2tf(A,B,C,D)

num =

         0    0.0000   25.0400    5.0080


den =

    1.0000    5.0325   25.1026    5.0080

%*****El mismo resultado se puede obtener introduciendo la siguiente orden*****

[num,den]=ss2tf(A,B,C,D,1)

num =

         0    0.0000   25.0400    5.0080


den =

    1.0000    5.0325   25.1026    5.0080
```

EJEMPLO 3-13

Sea un sistema con entradas y salidas múltiples. Cuando el sistema tiene más de una salida, la orden

[NUM, den] = ss2tf(A, B, C, D, iu)

produce las funciones de transferencia para todas las salidas para cada entrada. (Los coeficientes del numerador se devuelven en la matriz NUM que tiene tantas filas como salidas hay).

Sea el sistema definido por:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Este sistema tiene dos entradas y dos salidas. Se obtienen las cuatro funciones de transferencia siguientes: $Y_1(s)/U_1(s)$, $Y_2(s)/U_1(s)$, $Y_1(s)/U_2(s)$ y $Y_2(s)/U_2(s)$. (Cuando se considera la entrada u_1, suponemos que la entrada u_2 es cero y viceversa). Vea la siguiente salida que produce MATLAB

```
A=[0 1; -25 -4];
B=[1 1; 0 1];
C=[1 0; 0 1];
D=[0 0; 0 0];
[NUM,den]=ss2tf(A,B,C,D,1)

NUM =

     0     1     4
     0     0   -25

den =

     1     4    25

[NUM,den]=ss2tf(A,B,C,D,2)

NUM =

          0    1.0000    5.0000
          0    1.0000  -25.0000

den =

     1     4    25
```

Ésta es la representación en MATLAB de las cuatro funciones de transferencia siguientes:

$$\frac{Y_1(s)}{U_1(s)} = \frac{s+4}{s^2+4s+25}, \quad \frac{Y_2(s)}{U_1(s)} = \frac{-25}{s^2+4s+25}$$

$$\frac{Y_1(s)}{U_2(s)} = \frac{s+5}{s^2+4s+25}, \quad \frac{Y_2(s)}{U_2(s)} = \frac{s-25}{s^2+4s+25}$$

Capítulo 4 Análisis de la respuesta transitoria de sistemas discretos

4.1 INTRODUCCIÓN

Este capítulo estudia la respuesta transitoria de los sistemas discretos. La orden para la respuesta transitoria es diferente de la utilizada en el caso continuo. Para sistemas discretos, la orden más utilizada para la respuesta transitoria es

```
y = filter(num,den,x)
```

donde x es la entrada e y es la salida filtrada.

Antes de obtener los diagramas de las respuestas transitorias para sistemas discretos, resumiremos cómo generar las funciones de entrada para sistemas discretos. Las funciones de entrada que se van a considerar son la función delta de Kronecker (la cual se corresponde con la función de impulso unitario para sistemas en tiempo continuo), la función escalón, la función rampa, la función aceleración y la función de entrada arbitraria.

Generación de funciones de entrada

Entrada delta de Kronecker

La función delta de Kronecker se define por

$$u(0) = 1$$
$$u(k) = 0, \quad \text{para } k = 1, 2, 3, \ldots$$

La siguiente entrada delta de Kronecker

$$u(0) = 1$$
$$u(k) = 0, \quad \text{para } k = 1, 2, 3, \ldots, 60$$

se puede introducir en el programa MATLAB como

```
u = [1   zeros(1,60)]
```

Una entrada delta de Kronecker de magnitud 8 como

$$u(0) = 8$$
$$u(k) = 0, \quad \text{para } k = 1, 2, 3, \ldots, 40$$

se puede introducir en el programa como

```
u = [8   zeros(1,40)]
```

Entrada escalón

Una entrada escalón unitario como

$$u(k) = 1(k) = 1, \quad \text{para } k = 0, 1, 2, ..., 100$$

se puede introducir en el programa MATLAB como

```
u = ones(1,101)
```

o

```
u = [1   ones(1,100)]
```

Análogamente, una entrada escalón de magnitud 5, o

$$u(k) = 5*1(k) = 5, \quad \text{para } k = 0, 1, 2, ..., 50$$

se puede introducir en el programa MATLAB como

```
u = 5*ones(1,51)
```

o

```
.u = [5   5*ones(1,50)]
```

Entrada rampa

La entrada rampa unitaria se define por

$$u = t, \quad \text{para } 0 \leq t$$

Para sistemas discretos, $t = kT$, donde T es el periodo de muestreo (seg). Por consiguiente, la entrada rampa se puede escribir como

$$u(k) = kT, \quad \text{para } k = 0, 1, 2, ...$$

Si la rampa viene dada por

$$u(k) = kT, \quad \text{para } k = 0, 1, 2, ..., 50$$

entonces utilizaremos una de las siguientes formas:

```
u = 0:T:50*T      (T = periodo de muestreo, seg)
```

o

```
k = 0:50;     u = [k*T]
```

Es decir, si $T = 0.2$ seg y $k = 50$, se utiliza

```
u = 0:0.2:10
```

o

```
k = 0:50;     u = [0.2*k]
```

Entrada aceleración

Antes de considerar la entrada aceleración, consideremos la siguiente señal:

$$h = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6]$$

El cuadrado de esta señal en cada instante de muestreo viene dada por

$$w = [0 \quad 1 \quad 4 \quad 9 \quad 16 \quad 25 \quad 36]$$

La señal w se puede generar en MATLAB como sigue:

```
h=0:6

h =

     0     1     2     3     4     5     6

w=h.^2

w =

     0     1     4     9    16    25    36
```

Nótese que es necesario introducir un punto después de h para obtener el cuadrado del valor muestreado.

Ahora consideramos la entrada aceleración en general. Supongamos que la entrada viene dada por

$$u(k) = \frac{1}{2}(kT)^2, \quad \text{para } k = 0, 1, 2, \ldots$$

donde T es el periodo de muestreo. Asumamos, por ejemplo, que $T = 0.2$ seg, o

$$u(0) = 0$$

$$u(1) = \frac{1}{2}T^2 = \frac{1}{2}(0{,}2)^2$$

$$u(2) = \frac{1}{2}(2T)^2 = \frac{1}{2}(0{,}4)^2$$

$$u(k) = \frac{1}{2}(kT)^2 = \frac{1}{2}(0{,}2k)^2, \quad \text{para } k = 3, 4, 5, \ldots$$

Esta entrada aceleración se puede introducir en el programa MATLAB como sigue: Si $k = 0, 1, 2, 3, 4$, podemos introducir las siguientes órdenes:

```
k = 0:4;    u = [0.5*(0.2*k).^2]
```

La pantalla mostrará lo siguiente:

```
k=0:4;   u=[0.5*(0.2*k).^2]

u =

         0    0.0200    0.0800    0.1800    0.3200
```

Podemos introducir también las siguientes órdenes:

k = 0:4; m = [0.5*(0.2*k)'*(0.2*k)]; u = [diag(m)]'

La pantalla del ordenador mostrará lo siguiente:

```
k=0:4;   m=[0.5*(0.2*k)'*(0.2*k)]

m =

       0         0         0         0
0
         0    0.0200    0.0400    0.0600    0.0800
         0    0.0400    0.0800    0.1200    0.1600
         0    0.0600    0.1200    0.1800    0.2400
         0    0.0800    0.1600    0.2400    0.3200

u=[diag(m)]'

u =

         0    0.0200    0.0800    0.1800    0.3200
```

Resumiendo, si k varía de 0 a 40, podemos utilizar una de las siguientes formas:

a. k = 0:40; u = [0.5*(0.2*k).^2]
b. k = 0:40; u = [diag(0.5*(0.2*k)'*(0.2*k))]'

Entrada arbitraria

Si una entrada arbitraria se especifica como

$$u(0) = 3$$
$$u(1) = 2.5$$
$$u(2) = 1.2$$
$$u(k) = 0, \quad \text{para } k = 3, 4, 5, \ldots 80$$

la siguiente forma puede ser utilizada como la entrada:

u = [3 2.5 1.2 zeros(1,78)]

Índice del capítulo

El índice del capítulo es el siguiente: La sección 4-1 ha presentado el material introductorio. La sección 4-2 describe brevemente los filtros digitales. La sección 4-3 trata la respuesta de sistemas discretos a una entrada delta de Kronecker. La sección 4-4 muestra la respuesta a entradas escalón y a entradas arbitrarias. La sección 4-5 presenta la respuesta frente a rampa. La sección 4-6 discute las características de la respuesta transitoria de un sistema digital controlado con un PID y de sistemas de control deadbeat. Finalmente, la sección 4-7 presenta la transformación de modelos de sistemas continuos a discretos.

4.2 FILTROS DIGITALES

Considere un filtro digital cuya función de transferencia es

$$\frac{Y(z)}{X(z)} = \frac{b(z)}{a(z)}$$

donde $b(z)$ es el polinomio numerador en z y $a(z)$ es el polinomio denominador en z. La orden

y = filter(b,a,x) o y = filter(num,den,x)

filtra los datos del vector x con el filtro descrito por los vectores b y a (vectores num y den), creando el dato filtrado y. Considere la respuesta del siguiente filtro digital a una entrada x.

$$\frac{Y(z)}{X(z)} = \frac{10z + 2}{10z^3 + 5z^2 + 2z + 1} = \frac{10z^{-2} + 2z^{-3}}{10 + 5z^{-1} + 2z^{-2} + z^{-3}}$$

Nótese que los coeficientes del filtro están ordenados en potencias decrecientes de z o en potencias crecientes de z^{-1}.

Las dinámicas del sistema son descritas en términos de b y a (o en términos de num y den). Así las dinámicas del sistema son introducidas en el MATLAB como

```
b = [0  0  10  2];
a = [10  5  2  1];
```

o

```
num = [0  0  10  2];
den = [10  5  2  1];
```

Como el polinomio numerador es de menor grado que el polinomio denominador, primero debe ser rellenado con ceros por delante. (El polinomio numerador tiene que ser del mismo tamaño que el polinomio denominador).

Una vez que el numerador y el denominador han sido introducidos en MATLAB, introduzca la entrada x, la orden filter y la orden plot. Después, la respuesta a la entrada x aparecerá en la pantalla. En las secciones 4-3 a 4-5 se verán las respuestas de filtros digitales a diferentes señales de entrada.

Algunas consideraciones sobre filtros digitales

Nótese que en $Y(z)/X(z)$, donde

$$\frac{Y(z)}{X(z)} = \frac{b(1) + b(2)z^{-1} + ... + b(n+1)z^{-n}}{a(1) + a(2)z^{-1} + ... + a(n+1)z^{-n}}$$

si a(1)≠1, los coeficientes del filtro deben ser normalizados dividiendo todos los coeficientes entre a(1). Si a(1) = 0, la orden filter produce el siguiente mensaje de error:

El primer coeficiente del denominador del filtro debe ser distinto de cero.

Si este tipo de mensaje de error aparece, hacer las correcciones necesarias.

4.3 RESPUESTA A LA ENTRADA DELTA DE KRONECKER

Considere el siguiente sistema de control discreto:

$$\frac{Y(z)}{X(z)} = \frac{0{,}4673z^{-1} - 0{,}3393z^{-2}}{1 - 1{,}5327z^{-1} + 0{,}6607z^{-2}}$$
$$= \frac{0{,}4673z - 0{,}3393}{z^2 - 1{,}5327z + 0{,}6607}$$

Consideraremos la respuesta $y(k)$ de este sistema a una entrada delta de Kronecker.

Entrada delta de Kronecker

La entrada delta de Kronecker se define como

$$x(k) = 1, \quad \text{para } k = 0$$
$$= 0, \quad \text{para k} \neq 0$$

La transformada z de la entrada delta de Kronecker es

$$X(z) = 1$$

La entrada $x(k)$ en el programa de MATLAB se puede escribir como

```
x = [1  zeros(1,N)]
```

donde N corresponde con el final de la duración del proceso discreto considerado.

Cálculo de la transformada inversa *z* de *G(z)*

Calcular la transformada z de $G(z)$ es lo mismo que calcular la respuesta de $G(z)$ a una entrada delta de Kronecker. Para el sistema que estamos considerando, tenemos

$$\frac{Y(z)}{X(z)} = G(z) = \frac{0{,}4673z^{-1} - 0{,}3393z^{-2}}{1 - 1{,}5327z^{-1} + 0{,}6607z^{-2}}$$

Puesto que la transformada z de la entrada delta de Kronecker $X(z)$ es igual a la unidad, la respuesta del sistema a esta entrada es

$$Y(z) = G(z) = \frac{0{,}4673z^{-1} - 0{,}3393z^{-2}}{1 - 1{,}5327z^{-1} + 0{,}6607z^{-2}}$$

Por lo tanto la transformada inversa z de $G(z)$ es $y(0)$, $y(1)$, $y(2)$, ...
Para obtener la transformada inversa z de $G(z)$, se procede de la siguiente manera:
Introduzca el numerador y el denominador como sigue:

```
num = [0  0.4673  -0.3393]
den = [1  -1.5327  0.6607]
```

Introduzca la entrada delta de Kronecker.

```
x = [1  zeros(1,40)]
```

Después introduzca la orden

```
y = filter(num,den,x)
```

para obtener la respuesta $y(k)$ de $k = 0$ a $k = 40$.

Resumiendo, el programa en MATLAB para obtener la transformada inversa z o la respuesta a una entrada delta de Kronecker se muestra en en el programa 4-1. En la pantalla se mostrará la salida $y(k)$ desde $k = 0$ a 40 como sigue:

```
y =

  Columns 1 through 7

         0    0.4673    0.3769    0.2690    0.1632    0.0725    0.0032

  Columns 8 through 14

   -0.0429   -0.0679   -0.0758   -0.0712   -0.0591   -0.0436   -0.0277

  Columns 15 through 21

   -0.0137   -0.0027    0.0050    0.0094    0.0111    0.0108    0.0092

  Columns 22 through 28

    0.0070    0.0046    0.0025    0.0007   -0.0005   -0.0013   -0.0016

  Columns 29 through 35

   -0.0016   -0.0014   -0.0011   -0.0008   -0.0004   -0.0002    0.0000

  Columns 36 through 41

    0.0002    0.0002    0.0002    0.0002    0.0002    0.0001
```

Programa en MATLAB 4-1

```
%-------Calcular la transformada inversa z-------

%*****Calcular la transformada inversa z de G(z) es lo mismo que determinar la
%respuesta del sistema Y(z)/X(z)=G(z) a una entrada delta de Kronecker*****

%*****La respuesta a un impulso unitario se obtiene como la respuesta escalón
%unitario de sG(s)*****

%*****Introduzca el numerador y el denominador de G(z)*****

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];

%*****Introduzca la entrada delta de Kronecker x y la orden filter
%y=filter(num,den,x)*****

x=[1 zeros(1,40)];
y=filter(num,den,x)
```

(Nótese que los cálculos de MATLAB empiezan en la columna 1 y finalizan en la columna 41, y no van desde la columna 0 a la 40). Estos valores son la transformada inversa z de $G(z)$. Es decir,

$$
\begin{aligned}
y(1) &= 0.4673 \\
y(2) &= 0.3769 \\
y(3) &= 0.2690 \\
&\vdots \\
y(40) &= 0.0001
\end{aligned}
$$

Para representar los valores de la transformada inversa z de $G(z)$, se procede como sigue. Puesto que hemos elegido $0 \leq k \leq N = 40$, y el rango de la respuesta $y(k)$ se estima que se encuentre entre -1 y 1 (si esta estimación no es satisfactoria, cambie el rango después de una prueba), introduzca los rangos para el eje x ($0 \leq x \leq 40$) y el eje y ($-1 \leq y \leq 1$) de la siguiente manera:

```
v = [0  40  -1  1]
axis(v)
```

o combine las dos líneas de programa en una sola:

```
axis([0  40  -1  1])
```

Ahora añada un punto y coma al final de la línea

```
y = filter(num,den,x);
```

e introduzca

```
plot(y,'o')
```

representará la respuesta $y(k)$ frente a $k + 1$. Nótese que la gráfica en MATLAB comienza en $k = 1$ y acaba en $k = N + 1$. (Más adelante se mostrará cómo mover la gráfica un periodo de muestreo para que comience en $k = 0$).

Para obtener la respuesta $y(k)$ frente a $k + 1$, junto con la rejilla, el título de la gráfica, la etiqueta del eje x y la etiqueta del eje y, ejecute el programa de MATLAB 4-2. El resultado se muestra en la Figura 4-1.

Si se desea representar la respuesta $y(k)$ frente a $k + 1$ utilizando un trazo continuo, sustituya la orden

```
plot(y,'o')
```

por

```
plot(y,'-')
```

El resultado se muestra en la Figura 4-2.

Los vectores en MATLAB van desde 1 a $N + 1$ en lugar de 0 a N. Si se desea representar la respuesta $y(k)$ frente a k, en lugar de representar $y(k)$ frente a $k + 1$, necesitamos añadir la siguiente declaración

```
k = 0:40;
```

y cambiar la orden plot como sigue:

```
plot(k,y,'o')
```

Programa en MATLAB 4-2

```
%-------Respuesta a una entrada delta de Kronecker-------

%*****Calcular la respuesta del sistema Y(z)/X(z) pra una entrada delta de
%Kronecker*****

%*****Introduzca el numerador y el denominador de Y(z)/X(z)*****

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];

%*****Introduzca la entrada delta de Kronecker x, axis(v), la orden filter, y
la orden plot*****

x=[1 zeros(1,40)];
v=[0 40 -1 1];
axis(v);
y=filter(num,den,x);
plot(y,'o')

%*****Introduzca la rejilla, el tÌtulo de la gr·fica y las etiquetas de los ejes
%x e y*****

grid
title('Respuesta a una entrada delta de Kronecker')
xlabel('k+1')
ylabel('y(k)')
```

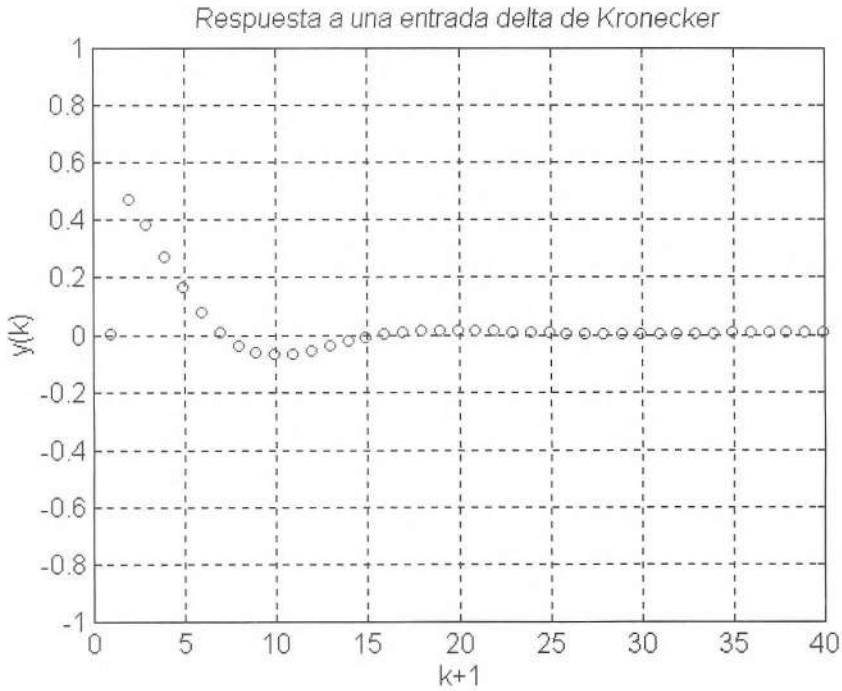


Figura 4-1

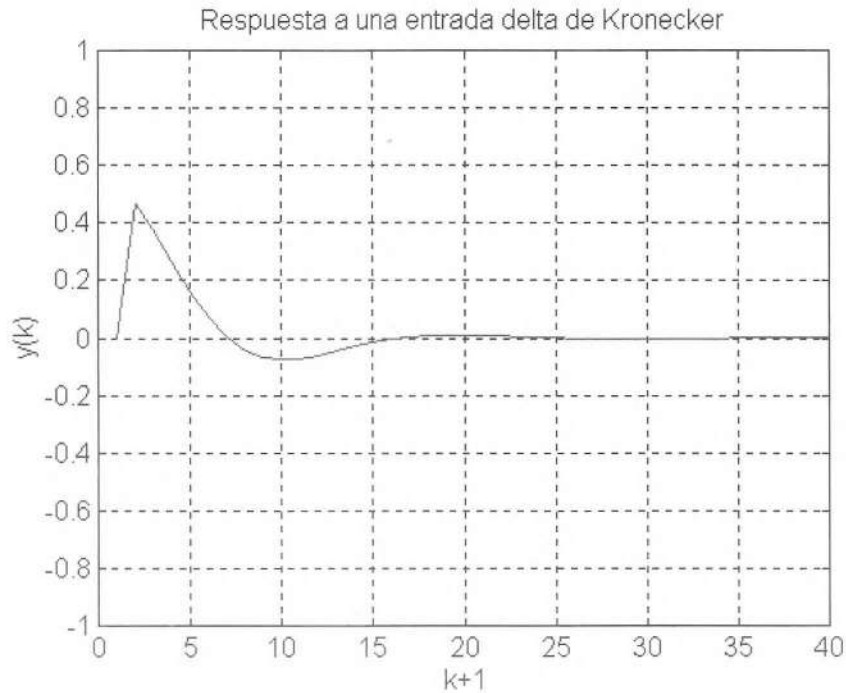


Figura 4-2

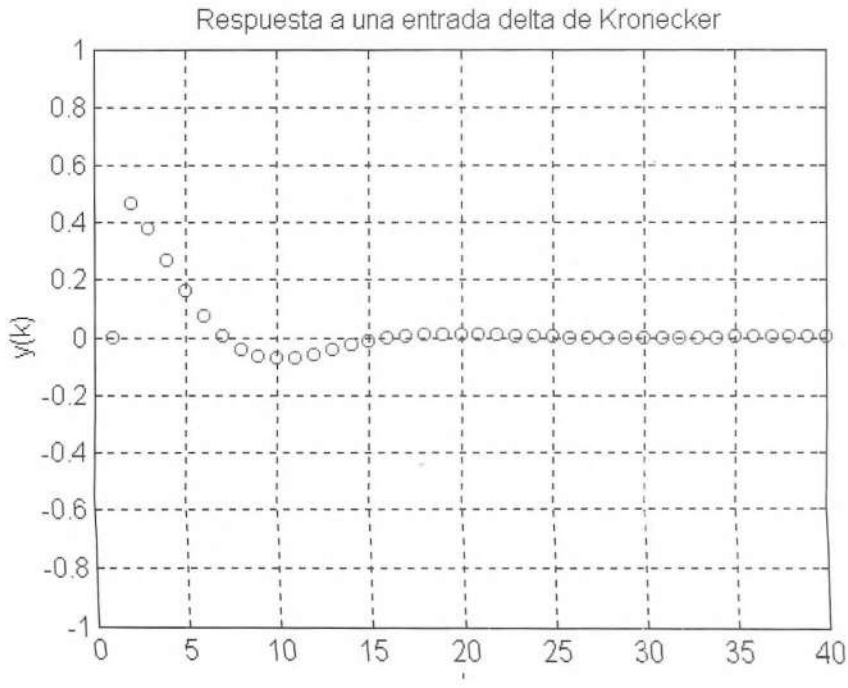


Figura 4-3

La gráfica resultante se muestra en la Figura 4-3.

Es importante darse cuenta que el número de elementos del vector fila k (k = 0, 1, 2, ..., 40) debe coincidir con la dimensión del vector del eje horizontal (x = 0, 1, 2, ..., 40). El programa modificado se encuentra en el programa de MATLAB 4-3.

Programa en MATLAB 4-3

```
%-------Respuesta a una entrada delta de Kronecker-

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];
x=[1 zeros(1,40)];
v=[0 40 -1 1];
axis(v);
k=0:40;
y=filter(num,den,x);
plot(y,'o')
grid
title('Respuesta a una entrada delta de Kronecker')
xlabel('k')
ylabel('y(k)')
```

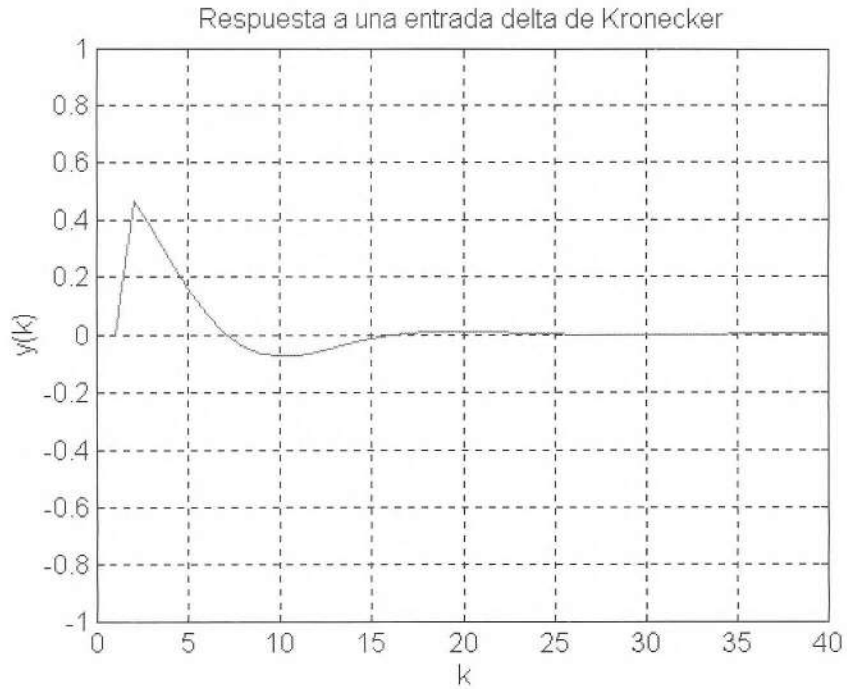


Figura 4-4

Si sustituimos la orden plot

```
plot(y,'o')
```

por

```
plot(k,y,'-')
```

obtenemos la gráfica mostrada en la Figura 4-4.

Si deseamos conectar puntos consecutivos (círculos abiertos,'o') con líneas rectas, necesitamos modificar la orden plot como sigue:

```
plot(k,y,'o',k,y,'-')
```

Una gráfica de $y(k)$ con círculos abiertos ('o') comienza en $k = 0$ y finaliza en $k = 40$. Análogamente,una gráfica de $y(k)$ con líneas continuas comienza en $k = 0$ y finaliza en $k = 40$. Véase el programa de MATLAB 4-4 y la gráfica resultante en la Figura 4-5.

Si deseamos que la gráfica comience en $k = 5$, es decir, obtener $y(k)$ frente a $k + 5$, necesitamos cambiar

```
k = 0:40;
```

a

```
k = 5:40;
```

El programa de MATLAB 4-5 nos dará la gráfica deseada, la cual se muestra en la Figura 4-6.

Programa en MATLAB 4-4

```
%-------Respuesta a una entrada delta de Kronecker-------

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];
x=[1 zeros(1,40)];
v=[0 40 -1 1];
axis(v);
k=0:40;
y=filter(num,den,x);
plot(k,y,'o',k,y,'-')
grid
title('Respuesta a una entrada delta de Kronecker')
xlabel('k')
ylabel('y(k)')
```

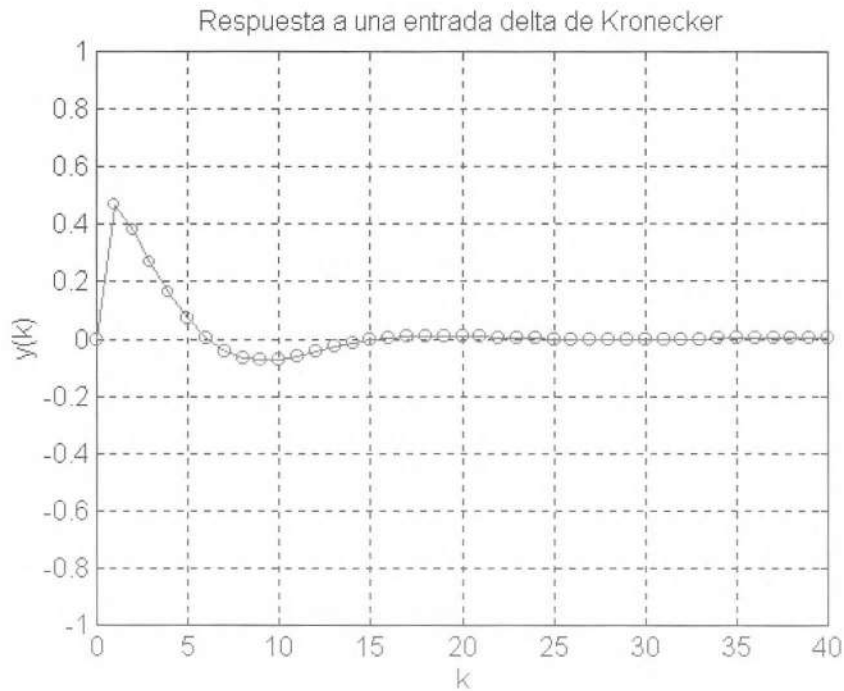


Figura 4-5

Programa en MATLAB 4-5

```
%-------Respuesta a una entrada delta de Kronecker-------

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];
x=[1 zeros(1,35)];
v=[0 40 -1 1];
axis(v);
k=5:40;
y=filter(num,den,x);
plot(k,y,'o',k,y,'-')
grid
title('Respuesta a una entrada delta de Kronecker')
xlabel('k+5')
ylabel('y(k)')
```

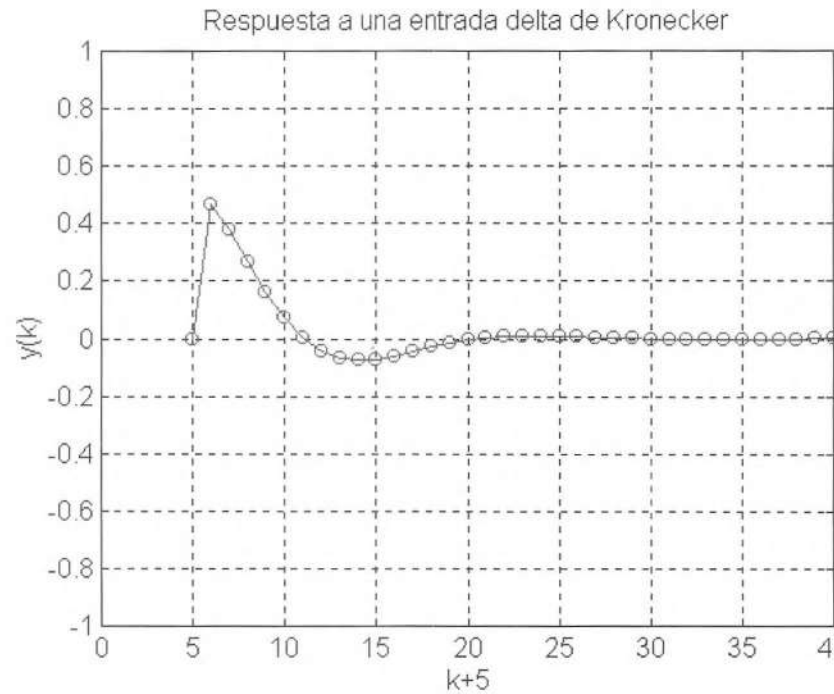


Figura 4-6

EJEMPLO 4-1

Obtener la transformada inversa z de la siguiente función de transferencia.

$$\frac{Y(z)}{X(z)} = G(z) = \frac{0{,}01409z^3 + 0{,}02818z^2 + 0{,}01409z}{z^3 - 2{,}7624z^2 + 2{,}5811z - 0{,}8187}$$

La transformada inversa z de $G(z)$ es lo mismo que la respuesta del sistema a una entrada delta de Kronecker. Vamos a obtener la transformada inversa z de $y(k)$ hasta $k = 40$.

Se introduce la siguiente entrada delta de Kronecker dentro del programa.

```
x = [1  zeros(1,40)]
```

Un programa en MATLAB para resolver este problema se muestra en el programa de MATLAB 4-6.

Programa en MATLAB 4-6

```
%-------Transformada inversa z-------

%*****La transformada inversa z se puede obtener calculando la respuesta del
%sistema Y(z)/X(z)para una entrada delta de Kronecker*****

%*****Introduzca el numerador y el denominador de Y(z)/X(z)*****

b=[0.01409 0.02818 0.01409 0];
a=[1 -2.7624 2.5811 -0.8187];

%*****Introduzca la entrada delta de Kronecker x y la orden filter*****

x=[1 zeros(1,40)];
y=filter(b,a,x)
```

A continuación se muestra la salida de la orden filter. Nótese que el vector de salida de MATLAB va desde la columna 1 hasta la columna 41. La columna 1 corresponde a $k = 0$ y la columna 41 a $k = 40$.

```
y =
  Columns 1 through 7
    0.0141    0.0671    0.1631    0.2888    0.4319    0.5811    0.7268

  Columns 8 through 14
    0.8616    0.9798    1.0778    1.1537    1.2072    1.2394    1.2524

  Columns 15 through 21
    1.2488    1.2320    1.2052    1.1718    1.1348    1.0970    1.0606

  Columns 22 through 28
    1.0275    0.9989    0.9756    0.9580    0.9460    0.9392    0.9372

  Columns 29 through 35
    0.9391    0.9442    0.9516    0.9604    0.9699    0.9794    0.9884

  Columns 36 through 41
    0.9965    1.0034    1.0089    1.0130    1.0156    1.0170
```

La transformada inversa z viene dada por:

$$
\begin{aligned}
y(0) &= 0.0141 \\
y(1) &= 0.0671 \\
y(2) &= 0.1631 \\
y(3) &= 0.2888 \\
&\vdots \\
y(40) &= 1.0170
\end{aligned}
$$

Para representar la transformada inversa z de $y(k)$ frente a k, modificar este programa de MATLAB como se muestra en el programa de MATLAB 4-7. La gráfica resultante de $y(k)$ se muestra en la Figura 4-7.

Programa en MATLAB 4-7

```
%------Representación de la transformada inversa z de y(k) respecto de k------

b=[0.01409 0.02818 0.01409 0];
a=[1 -2.7624 2.5811 -0.8187];
x=[1 zeros(1,40)];
v=[0 40 0 1.4];
axis(v);
k=0:40;
y=filter(b,a,x);
plot(k,y,'o'), grid, title('Gráfica de la transformada inversa z de y(k)
respecto de k')
xlabel('k'), ylabel('y(k)')
```

Figura 4-7

EJEMPLO 4-2

Considere la ecuación diferencial

$$x\ =\ (k+2)\ =\ x(k+1)+x(k) \tag{4-1}$$

donde $x(0) = 0$ y $x(1) = 1$. Nótese que $x(2) = 1$, $x(3) = 2$, $x(4) = 3$, ... las series 0, 1, 1, 2, 3, 5, 8, 13, ... son conocidas como las series de Fibonacci. Obtener la serie de Fibonacci hasta $k = 30$.

La transformada z de la Ec.(4-1) es

$$z^2X(z) - z^2x(0) - zx(1)\ =\ zX(z) - zx(0) + X(z)$$

Resolviendo esta ecuación para $X(z)$ tenemos

$$X(z)\ =\ \frac{z^2x(0) + zx(1) - zx(0)}{z^2 - z - 1}$$

Sustituyendo el dato inicial $x(0) = 0$ y $x(1) = 1$ en esta última ecuación, tenemos

$$X(z)\ =\ \frac{z}{z^2 - z - 1}$$

La transformada inversa z de $X(z)$ dará las series de Fibonacci.

Para obtener la transformada inversa z de $X(z)$, obtener la respuesta de este sistema a una entrada delta de Kronecker. Véase el programa de MATLAB 4-8.

Programa en MATLAB 4-8

```
%-------Series de Fibonacci-------

%*****Las series de Fibonacci se pueden generar como la respuesta de X(z) a una
%entrada delta de Kronecker, donde X(z)=z/(z^2-z-1)*****

%*****Introduzca el numerador y el denominador de X(z)*****

num=[0 1 0];
den=[1 -1 -1];

%*****Introduzca la entrada delta de Kronecker y la orden filter*****

u=[1 zeros(1,30)];
x=filter(num,den,u)
```

La salida filtrada y mostrada a continuación da las series de Fibonacci.

```
x =

  Columns 1 through 6

         0         1         1         2         3         5

  Columns 7 through 12

         8        13        21        34        55        89

  Columns 13 through 18

       144       233       377       610       987      1597

  Columns 19 through 24

      2584      4181      6765     10946     17711     28657

  Columns 25 through 30

     46368     75025    121393    196418    317811    514229

  Column 31

    832040
```

Nótese que la columna 1 corresponde a k = 0 y la columna 31 corresponde a k = 30. La serie de Fibonacci está dada por

$$
\begin{aligned}
x(0) &= 0 \\
x(1) &= 1 \\
x(2) &= 1 \\
x(3) &= 2 \\
x(4) &= 3 \\
x(5) &= 5 \\
&\vdots \\
x(29) &= 514{,}229 \\
x(30) &= 832{,}040
\end{aligned}
$$

4.4 RESPUESTA A UNA ENTRADA ESCALÓN

Esta sección estudia las repuestas de sistemas discretos a entradas escalón y a entradas arbitrarias. Empezaremos con la respuesta a escalón de sistemas discretos.

Respuesta escalón

Utilizando un sistema de ejemplo, explicaremos los métodos para obtener las gráficas de la respuesta a escalón utilizando MATLAB.

Considere el sistema dado por la siguiente función de transferencia

$$\frac{Y(z)}{U(z)} = \frac{0{,}4673z^{-1} - 0{,}3393z^{-2}}{1 - 1{,}5327z^{-1} + 0{,}6607z^{-2}}$$

$$= \frac{0{,}4673z - 0{,}3393}{z^2 - 1{,}5327z + 0{,}6607}$$

donde

$$U(z) = \text{Z [unit step]}$$

Presentaremos dos métodos para obtener la respuesta a escalón.

Método 1: La entrada escalón unitario $u(k)$ se puede escribir como

$$u(k) = 1, \quad \text{para } k = 0, 1, 2, \ldots$$

Supongamos que queremos la respuesta hasta $k = 40$. Entonces la entrada u se puede escribir como

```
u = ones(1,41)
```

o

```
u = [1    ones(1,40)]
```

Un programa de MATLAB para representar la curva de la respuesta de un sistema a un escalón unitario se muestra en el programa de MATLAB 4-9. La gráfica resultante se muestra en la Figura 4-8. Nótese en esta gráfica que la respuesta comienza desde $k = 1$. (Como se explicó anteriormente, los vectores en MATLAB en este caso van desde $k = 1$ hasta $k = 41$. Si deseamos representar la respuesta $y(k)$ desde $k = 0$ hasta $k = 40$, necesitamos hacer una pequeña modificación en el programa de MATLAB. Véase el programa de MATLAB 4-10).

El programa de MATLAB 4-10 nos da la curva de la respuesta comenzando en $k = 0$. Véase la gráfica de la respuesta mostrada en la Figura 4-9.

Programa en MATLAB 4-9

```
%-------Respuesta a un escalón unitario--

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];
u=ones(1,41);
v=[0 40 0 1.6];
axis(v);
y=filter(num,den,u);
plot(y,'o')
grid
title('Respuesta a un escalón unitario')
xlabel('k+1')
ylabel('y(k)')
```

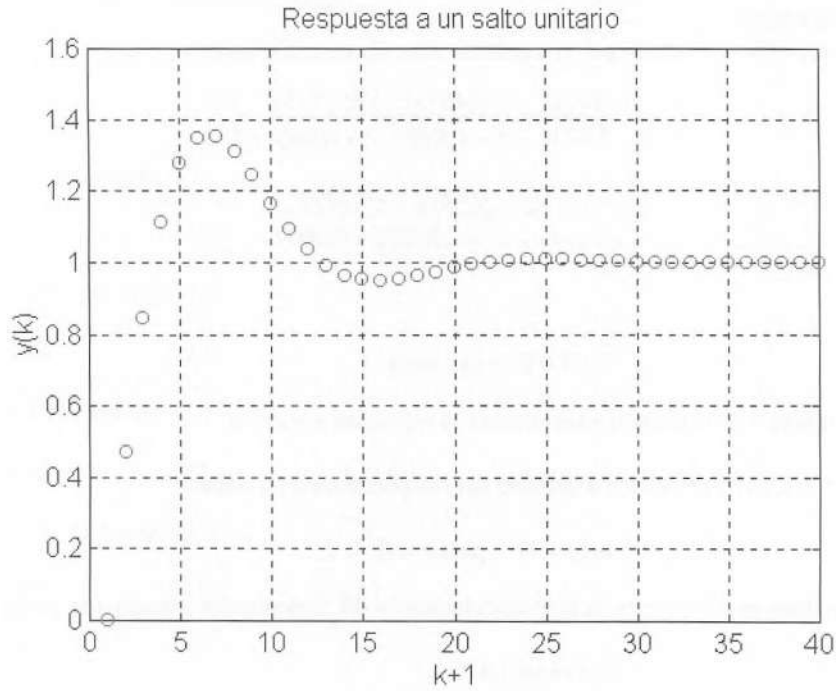


Figura 4-8

Programa en MATLAB 4-10

```
%-------Respuesta a un escalón unitario-------

num=[0 0.4673 -0.3393];
den=[1 -1.5327 0.6607];
u=ones(1,41);
v=[0 40 0 1.6];
axis(v);
k=0:40;
y=filter(num,den,u);
plot(k,y,'o')
grid
title('Respuesta a un escalón unitario')
xlabel('k')
ylabel('y(k)')
```

Método 2: en este método, multiplicamos $Y(z)/U(z)$ por $U(z)$, donde $U(z) = z/(z\text{-}1)$ para la entrada escalón unitario. Por lo tanto podemos obtener la transformada inversa z de $Y(z)$ mediante la respuesta a una entrada escalón unitario.

Considere el mismo sistema que se discutió anteriormente o

$$\frac{Y(z)}{U(z)} = \frac{0{,}4673z - 0{,}3393}{z^2 - 1{,}5327z + 0{,}6607}$$

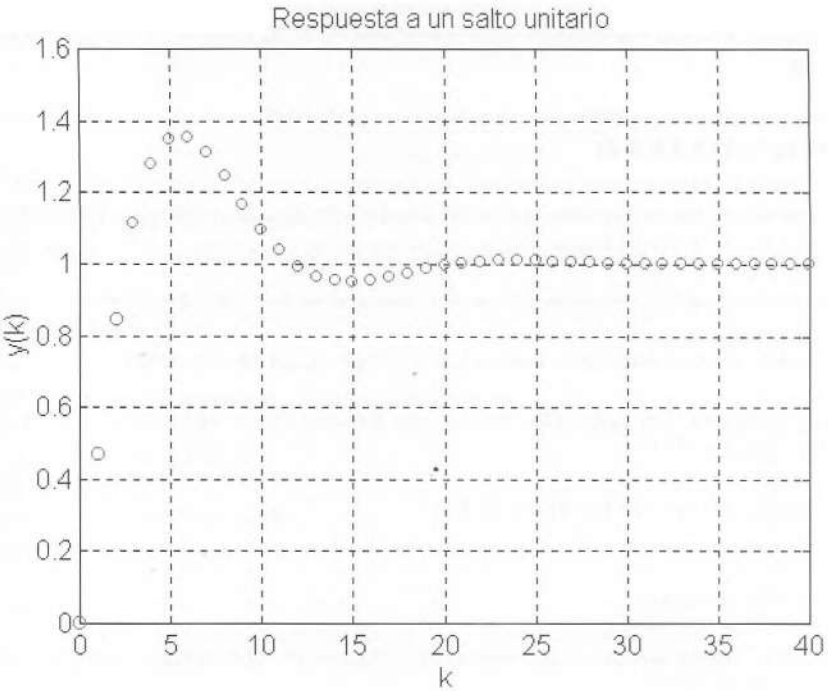


Figura 4-9

Multiplicando $U(z) = z/(z\text{-}1)$ a ambos lados de esta última ecuación, obtenemos

$$Y(z) = \frac{0{,}4673z - 0{,}3393}{z^2 - 1{,}5327z + 0{,}6607}U(z)$$

$$= \frac{0{,}4673z - 0{,}3393}{z^2 - 1{,}5327z + 0{,}6607}\ \frac{z}{z-1}$$

$$= \frac{0{,}4673z^2 - 0{,}3393z}{z^3 - 2{,}5327z^2 + 2{,}1934z - 0{,}6607}$$

Esta última ecuación puede ser escrita de la siguiente manera:

$$\frac{Y(z)}{X(z)} = \frac{0{,}4673z^2 - 0{,}3393z}{z^3 - 2{,}5327z^2 + 2{,}1934z - 0{,}6607}$$

donde $X(z) = 1$. Puesto que

$$X(z) = \sum_{k=0} x(k)z^{-k} = x(0) + x(1)z^{-1} + x(2)z^{-2} + \ldots = 1$$

tenemos

$x(0) = 1$
$x(\mathrm{k}) = 0$, para $k = 1, 2, 3, \ldots$

Si deseamos representar $y(k)$ desde $k = 0$ hasta 40, debemos escribir la entrada $x(k)$ de la siguiente forma:

$$x = [1 \quad \text{zeros}(1,40)]$$

Un programa en MATLAB para representar la respuesta a un escalón unitario utilizando la aproximación de la transformada inversa z se muestra en el programa de MATLAB 4-11. La gráfica resultante se muestra en la Figura 4-10.

Programa en MATLAB 4-11

```
%-------Respuesta a un escalón unitario obtenida como la transformada inversa z
%de G(z)=[Y(z)/X(z)][z/(z-1)]-------

%*****Introduzca el numerador y el denominador de G(z)*****

num=[0 0.4673 -0.3393 0]; den=[1 -2.5327 2.1934 -0.6607];

%*****Introduzca la entrada delta de Kronecker, axis(v), k, la orden filter y
%la orden plot*****

x=[1 zeros(1,40)]; v=[0 40 0 1.6];
axis(v);
k=0:40;
y=filter(num,den,x);
plot(k,y,'o',k,y,'-')
grid, title('Respuesta a un escalón unitario obtenida como la transformada
inversa z de G(z)')
xlabel('k'), ylabel('y(k)')
```

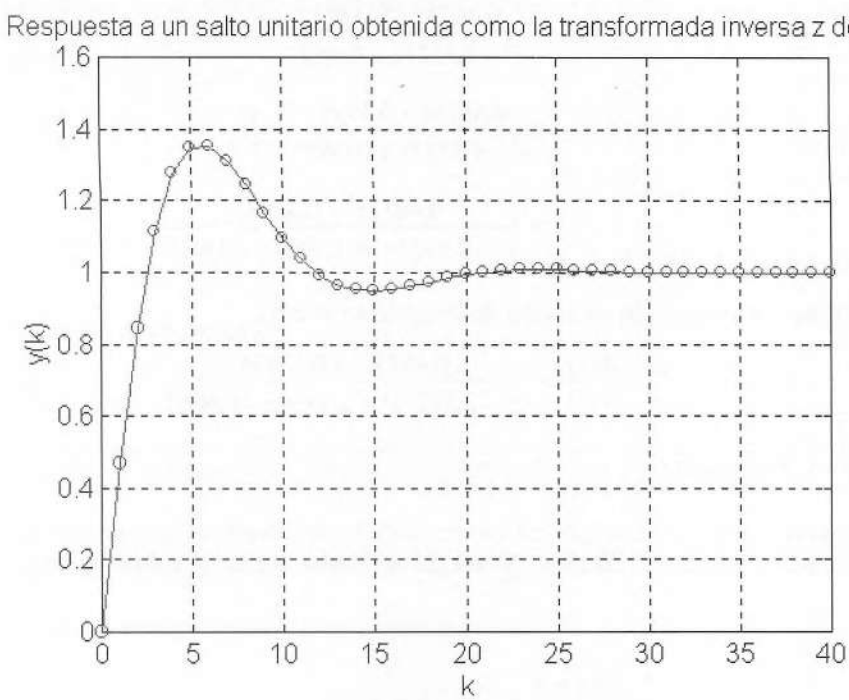


Figura 4-10

EJEMPLO 4-3

Representar la curva de la respuesta a un escalón unitario de un sistema cuya función de transferencia es:

$$\frac{Y(z)}{U(z)} = \frac{0{,}01409 + 0{,}02818z^{-1} + 0{,}01409z^{-2}}{1 - 1{,}7624z^{-1} + 0{,}8187z^{-2}}$$

$$= \frac{0{,}01409z^2 + 0{,}02818z + 0{,}01409}{z^2 - 1{,}7624z + 0{,}8187}$$

Un programa en MATLAB para representar la curva de la respuesta a un escalón unitario se muestra en el programa de MATLAB 4-12. La gráfica resultante se muestra en la Figura 4-11.

Programa en MATLAB 4-12

```
%-------Respuesta a un escalón unitario-------

b=[0.01409 0.02818 0.01409];
a=[1 -1.7624 0.8187];
u=ones(1,101); v=[0 100 0 1.4];
axis(v);
k=0:100;
y=filter(b,a,u);
plot(k,y,'o'), grid, title('Respuesta a un escalón unitario')
xlabel('k'), ylabel('y(k)')
```

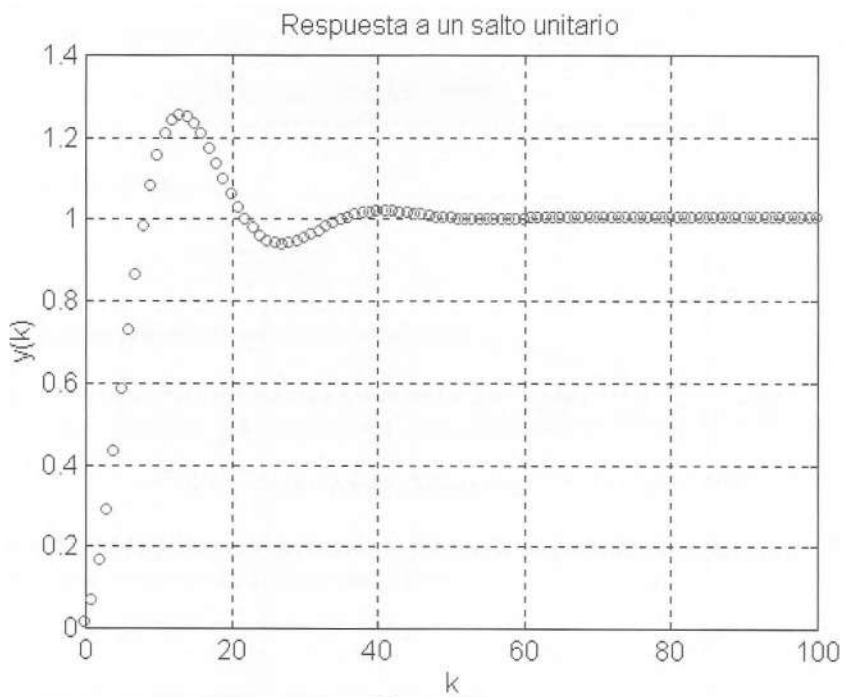


Figura 4-11

EJEMPLO 4-4

Representar la respuesta a un escalón unitario de la siguiente función de transferencia de un sistema en lazo cerrado. La función de transferencia en lazo cerrado es:

$$\frac{Y(z)}{U(z)} = \frac{0{,}3205z^{-3} - 0{,}1885z^{-4}}{1 - 1{,}3679z^{-1} + 0{,}3679z^{-2} + 0{,}3205z^{-3} - 0{,}1885z^{-4}}$$

$$= \frac{0{,}3205z - 0{,}1885}{z^4 - 1{,}3679z^3 + 0{,}3679z^2 + 0{,}3205z - 0{,}1885}$$

Un programa en MATLAB para representar la respuesta de un escalón unitario se muestra en el programa de MATLAB 4-13. La gráfica resultante se muestra en la Figura 4-12. Nótese que la respuesta a un escalón unitario alcanza exactamente el valor final (la unidad) con rapidez sin sobreelongación. Vamos a encontrar los polos en lazo cerrado del sistema.

Programa en MATLAB 4-13

```
%-------Respuesta a un escalón unitario-------

num=[0 0 0 0.3205 -0.1885]; den=[1 -1.3679 0.3679 0.3205 -0.1885];
u=[1 ones(1,50)];
v=[0 50 0 1.6]; axis(v);
k=0:50;
y=filter(num,den,u);
plot(k,y,'o'), grid, title('Respuesta a un escalón unitario'), xlabel('k'),
ylabel('y(k)')
```

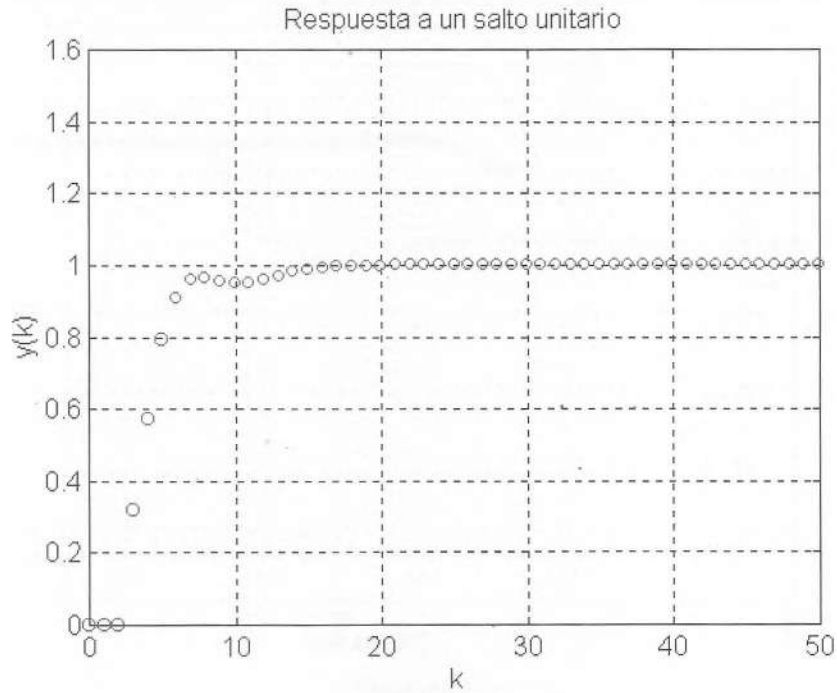


Figura 4-12

La ecuación característica para este sistema en lazo cerrado se obtiene del polinomio denominador como sigue:

$$\text{den}(z) = z^4 - 1.3679z^3 + 0.3679z^2 + 0.3205z - 0.1885 = 0$$

Para encontrar las raíces características, introduzca la orden

```
r = roots(den)
```

El resultado es el siguiente:

```
den =

    1.0000   -1.3679    0.3679    0.3205   -0.1885

r=roots(den)

r =

   0.5625+ 0.4095i
   0.5625- 0.4095i
   0.7572
  -0.5143
```

EJEMPLO 4-5

En este ejemplo, consideramos la respuesta del sistema a una entrada arbitraria. Supongamos que la función de transferencia de un sistema viene dada por

$$\frac{Y(z)}{U(z)} = \frac{0{,}3679z + 0{,}2642}{z - 1{,}3679z + 0{,}3679}$$

La entrada $u(k)$ del sistema es

$$u(0) = 1.582$$
$$u(1) = -0.582$$

$u(k) = 0$, para $k = 2, 3, 4, \ldots$

Representar la respuesta $y(k)$ del sistema a una entrada $u(k)$ para $k = 0, 1, 2, \ldots, 25$.
La función de entrada $u(k)$ se puede introducir en un programa MATLAB como

```
u = [1.582  -0.582  zeros(1,24)]
```

Un programa en MATLAB para representar la respuesta se muestra en el programa de MATLAB 4-14. Una gráfica de la respuesta se muestra en la Figura 4-13.

Programa en MATLAB 4-14

```
%-------Respuesta a una entrada arbitraria-------

%*****Respuesta del sistema a una entrada arbitraria especificada*****

num=[0 0.3679 0.2642];
den=[1 -1.3679 0.3679];

%*****Introduzca la entrada dada*****

u=[1.582 -0.5820 zeros(1,24)];

%*****Introduzca axis(v), k, la orden filter y la orden plot*****

v=[0 25 0 2];
axis(v);
k=0:25;
y=filter(num,den,u);
plot(k,y,'o')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid, title('Respuesta del sistema a una entrada arbitraria especificada')
xlabel('k'), ylabel('y(k)')
```

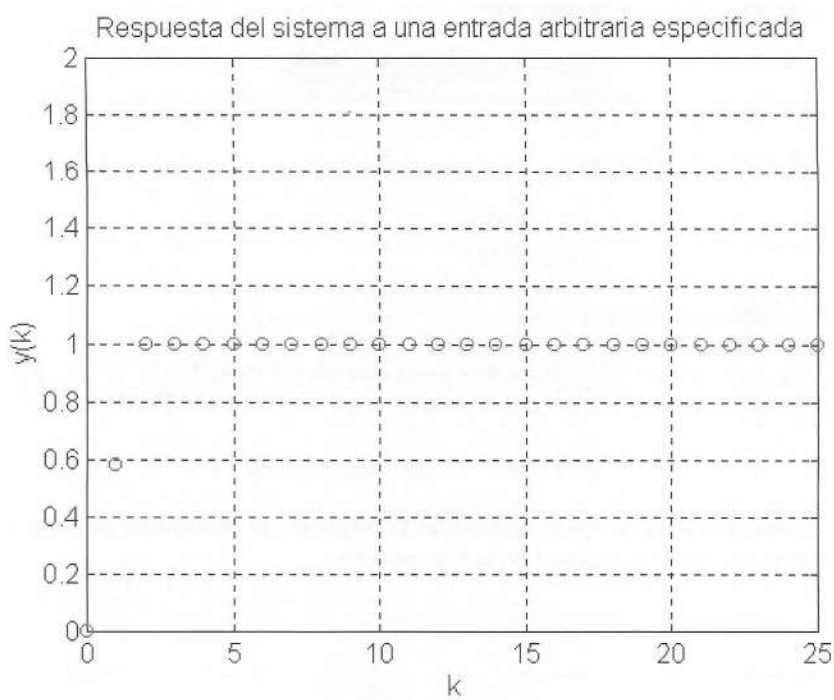


Figura 4-13

EJEMPLO 4-6

Considere el mismo sistema que en el Ejemplo 4-5. Suponga que que la entrada $u(k)$ viene dada por

$$u(0) = u(1) = u(2) = 1$$
$$u(3) = u(4) = u(5) = -1$$
$$u(k) = 0, \quad \text{para } k = 6, 7, 8, ..., 25$$

Encontrar la respuesta $y(k)$ a esta entrada y obtener la gráfica de $y(k)$ frente a k.
La función de entrada se puede escribir como

```
u = [1  1  1  -1  -1  -1  zeros(1,20)]
```

Un programa en MATLAB para representar la respuesta a esta entrada se muestra en el programa de MATLAB 4-15. La gráfica resultante se muestra en la Figura 4-14.

Programa en MATLAB 4-15

```
%-------Respuesta a una entrada arbitraria-------

%*****Introduzca el numerador y el denominador del sistema*****

num=[0 0.3679 0.2642];
den=[1 -1.3679 0.3679];

%*****Introduzca la entrada dada*****

u=[1 1 1 -1 -1 -1 zeros(1,20)];

%*****Introduzca axis(v), k, la orden filter y la orden plot*****

v=[0 25 -3 3];
axis(v);
k=0:25;
y=filter(num,den,u);
plot(k,y,'o')

%*****Introduzca la rejilla, el tÌtulo de la gr·fica y las etiquetas de los ejes
%x e y*****

grid
title('Respuesta del sistema a una entrada arbitraria especificada')
xlabel('k')
ylabel('y(k)')
```

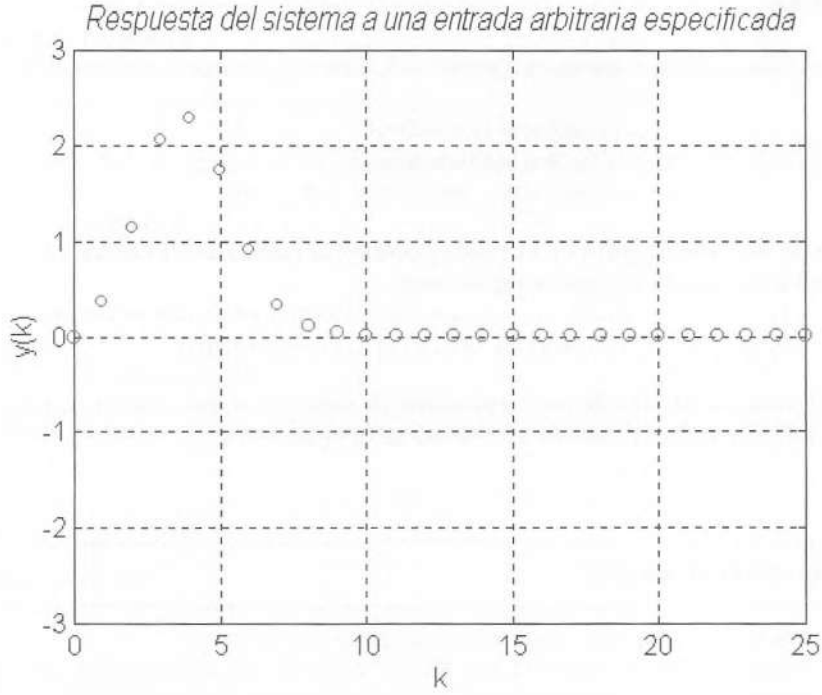


Figura 4-14

4.5 RESPUESTA A UNA ENTRADA EN RAMPA

En esta sección presentaremos la respuesta a una entrada en rampa unitaria. Para analizar la respuesta en rampa, es necesario especificar el periodo de muestreo T.

Sea el sistema descrito por

$$\frac{Y(z)}{U(z)} = \frac{0{,}7870z^{-1}}{1 - 0{,}8195z^{-1} + 0{,}6065z^{-2}}$$

$$= \frac{0{,}7870z}{z^2 - 0{,}8195z + 0{,}6065}$$

El periodo de muestreo T es de 0.5 seg. Representar la respuesta a una entrada en rampa unitaria hasta $k = 20$.

La respuesta a una entrada en rampa unitaria se puede obtener aplicando la entrada en rampa $u = kT$ ($k = 0, 1, 2, ...$) al sistema o multiplicando la entrada $U(z) = Tz/(z-1)^2$ por la función de transferencia discreta del sistema y utilizando como entrada la delta de Kronecker. Consideraremos ambos procedimientos.

Método 1: La respuesta a una entrada en rampa unitaria se obtiene aplicando dicha entrada al sistema.

La entrada en rampa unitaria puede expresarse mediante:

$$u = kT, \quad k = 0, 1, 2, ...$$

En el programa en MATLAB, esta entrada puede escribirse como:

```
k = 0:N;    u = [k*T];
```

donde N es el final del proceso considerado.

En la respuesta a una entrada en rampa, es importante especificar el periodo de muestreo T, ya que la pendiente de la entrada en rampa cuando se representa respecto de k es T. (Para T = 0.5 seg., la pendiente de la entrada en rampa unitaria cuando se representa respecto de k es 0.5).

En el Programa 4-16 en MATLAB se presenta cómo representar la respuesta a una entrada en rampa unitaria para el sistema considerado. En la Figura 4-15 se muestra la gráfica resultante.

Método 2: La respuesta a una entrada en rampa unitaria se obtiene como la transformada z inversa de $G(z) = [Y(z)/U(z)]U(z)$.

La transformada z de la entrada en rampa unitaria es:

$$U(z) = \frac{Tz^{-1}}{(1-z^{-1})^2} = \frac{Tz}{(z-1)^2}$$

Multiplicando $U(z)$ por $Y(z)/U(z)$, obtenemos

$$\frac{Y(z)}{U(z)}U(z) = G(z) = \frac{Y(z)}{U(z)}\frac{T(z)}{(z-1)^2}$$

Para el sistema considerado, $G(z)$ es

$$G(z) = Y(z) = \frac{0{,}7870z}{z^2 - 0{,}8195z + 0{,}6065}\frac{0{,}5z}{z^2 - 2z + 1}$$

$$= \frac{0{,}3935z^2}{z^4 - 2{,}8195z^3 + 3{,}2455z^2 - 2{,}0325z + 0{,}6065}$$

Programa en MATLAB 4-16

```
%-------Respuesta a una rampa unitaria-------

%*****Introduzca el numerador y el denominador de G(z)*****

num=[0 0.7870 0];
den=[1 -0.8195 0.6065];

%*****Introduzca la entrada delta de Kronecker, k, la orden filter y
%la orden plot*****

k=0:20;
u=[0.5*k];
y=filter(num,den,u);
plot(k,y,'o',k,y,'-',k,0.5*k,'--')

%*****Introduzca la rejilla, el tÌtulo de la gr·fica y las etiquetas de los ejes
%x e y*****

grid, title('Respuesta a una rampa unitaria')
xlabel('k')
ylabel('y(k)')
```

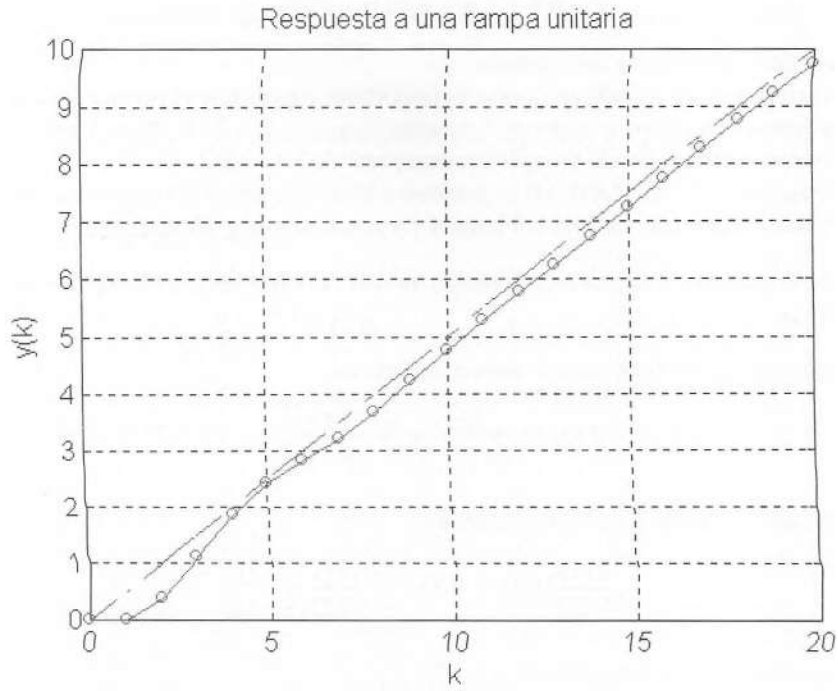


Figura 4-15

La transformada z inversa de $G(z)$ dará la respuesta a una entrada en rampa unitaria. El Programa 4-17 en MATLAB se puede utilizar para representar la respuesta a una entrada en rampa unitaria utilizando el método de la transformada z inversa. En la Figura 4-16 se muestra la gráfica resultante.

Programa en MATLAB 4-17

```
%-------Respuesta a una rampa unitaria-------

%*****La respuesta a una rampa unitariase obtiene como la transformada inversa
%z de G(z)=[Y(z)/U(z)][Tz/(z-1)^2]*****

%*****Introduzca el numerador y el denominador de G(z)*****

num=[0 0 0.3935 0 0]; den=[1 -2.8195 3.2455 -2.0325 0.6065];

%*****Introduzca la entrada delta de Kronecker, k, la orden filter y la orden
%plot*****

x=[1 zeros(1,20)]; k=0:20; y=filter(num,den,x);
plot(k,y,'o',k,y,'-',k,0.5*k,'--')

%*****Introduzca la rejilla, el tÌtulo de la gr·fica y las etiquetas de los ejes
%x e y*****

grid, title('Respuesta a una rampa unitaria obtenida como la transformada
inversa z')
xlabel('k')
ylabel('y(k)')
```

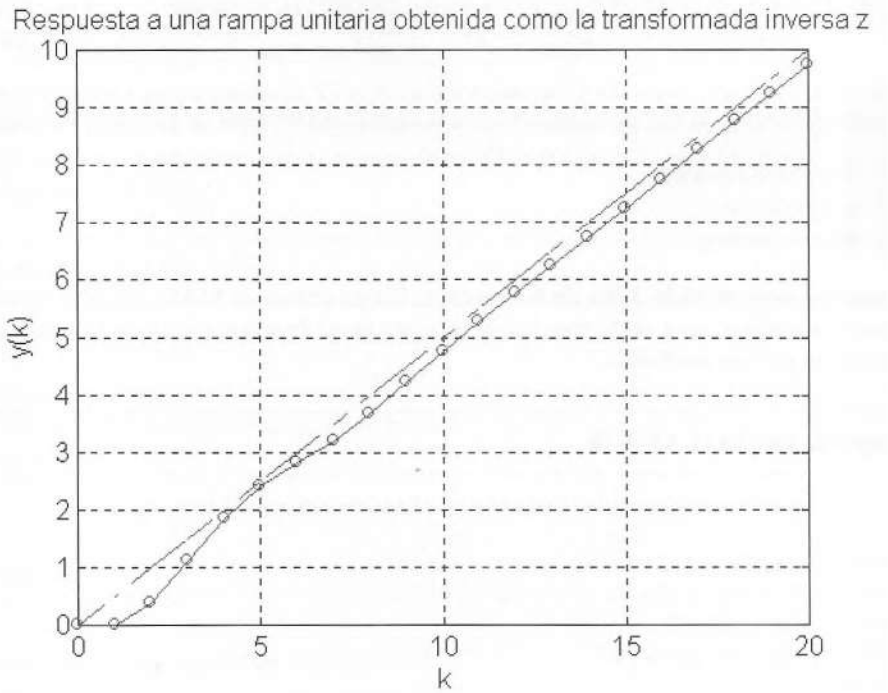


Figura 4-16

4.6 EJEMPLOS DE APLICACIÓN

En esta sección, en primer lugar estudiaremos las características de la respuesta transitoria del sistema de control digital PID que se muestra en la Figura 4-17. (Este sistema se ha diseñado para que presente error en estado estacionario nulo en respuesta a una entrada en rampa). En segundo lugar presentaremos la salida de los sistemas de respuesta plana.

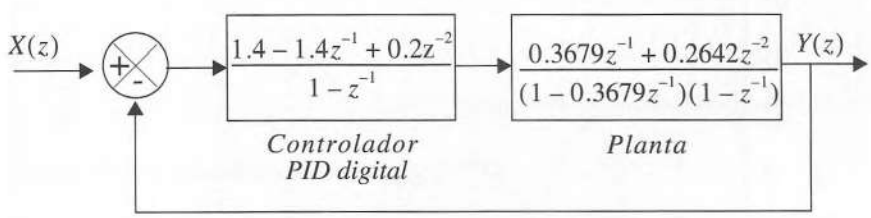


Figura 4-17

EJEMPLO 4-7

Sea el sistema de control digital que se muestra en la Figura 4-17. El periodo de muestreo T es 1 seg. La función de transferencia discreta para el sistema en lazo cerrado es:

$$\frac{Y(z)}{X(z)} = \frac{0{,}5151z^{-1} - 0{,}1452z^{-2} - 0{,}2963z^{-3} + 0{,}0528z^{-4}}{1 - 1{,}8528z^{-1} + 1{,}5906z^{-2} - 0{,}6642z^{-3} + 0{,}0528z^{-4}}$$

o

$$\frac{Y(z)}{X(z)} = \frac{0{,}5151z^3 - 0{,}1452z^2 - 0{,}2963z + 0{,}0528}{z^4 - 1{,}8528z^3 + 1{,}5906z^2 - 0{,}6642z + 0{,}0528}$$

Consideraremos la respuesta de este sistema para los tres tipos de entradas siguientes:

1. Delta de Kronecker
2. Salto unitario
3. Rampa unitaria

Respuesta a una entrada delta de Kronecker. Un programa en MATLAB para representar la respuesta del sistema a la entrada delta de Kronecker se muestra en el Programa 4-18 en MATLAB. En la Figura 4-18 se representa la gráfica resultante.

Programa en MATLAB 4-18

```
%-------Respuesta a una entrada delta de Kronecker-------

num=[0 0.5151 -0.1452 -0.2963 0.0528];
den=[1 -1.8528 1.5906 -0.6642 0.0528];
x=[1 zeros(1,40)];
v=[0 40 -1 1]; axis(v);
k=0:40;
y=filter(num,den,x);
plot(k,y,'o',k,y,'-'), grid, title('Respuesta a una entrada delta de Kro-
necker')
xlabel('k'), ylabel('y(k)')
```

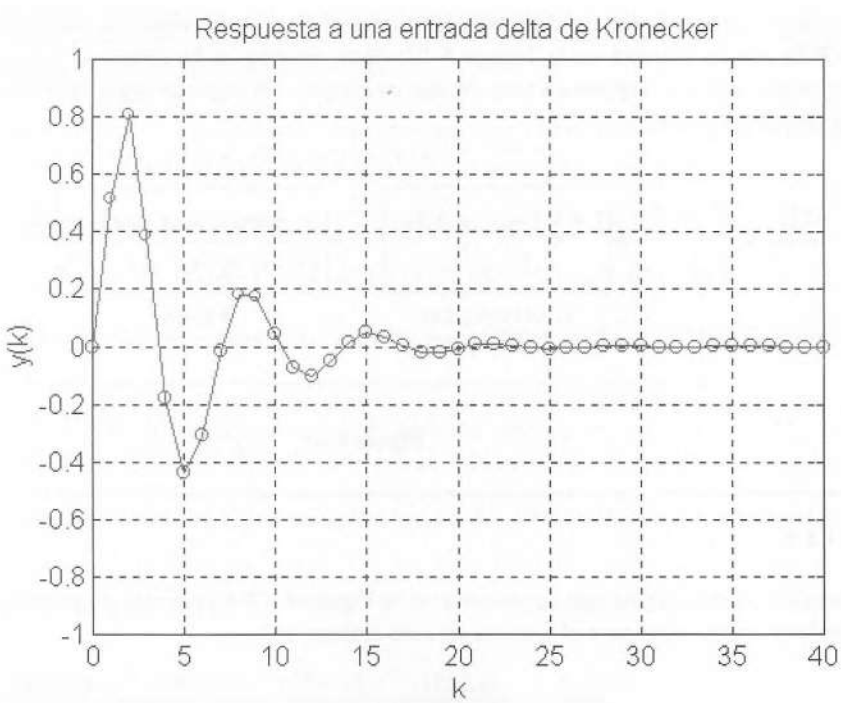


Figura 4-18

Respuesta a una entrada salto unitario. Una gráfica de la respuesta a un salto unitario del sistema se puede obtener introduciendo en el computador el Programa 4-19 en MATLAB. En la Figura 4-19 se representa la gráfica resultante.

Respuesta a una entrada rampa unitaria. El periodo de muestreo T es 1 seg. Por tanto la entrada en rampa es x = [k*T] = [k]. Un programa en MATLAB para representar la respuesta del sistema a una entrada en rampa unitaria se puede obtener introduciendo en el computador el Programa 4-20 en MATLAB. En la Figura 4-20 se representa la gráfica resultante.

Programa en MATLAB 4-19

```
%-------Respuesta a un escalón unitario-------

num=[0 0.5151 -0.1452 -0.2963 0.0528];
den=[1 -1.8528 1.5906 -0.6642 0.0528];
x=[1 ones(1,40)];
v=[0 40 0 2];
axis(v);
k=0:40;
y=filter(num,den,x);
plot(k,y,'o',k,y,'-')
grid
title('Respuesta a un escalón unitario')
xlabel('k')
ylabel('y(k)')
```

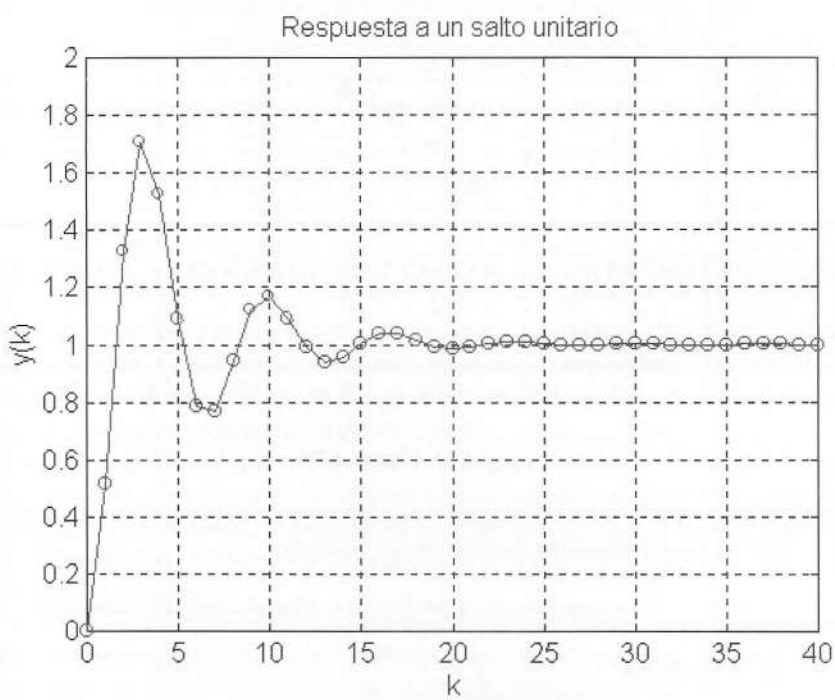


Figura 4-19

Programa en MATLAB 4-20

```
%-------Respuesta a una rampa unitaria-

num=[0 0.5151 -0.1452 -0.2963 0.0528];
den=[1 -1.8528 1.5906 -0.6642 0.0528];
v=[0 16 0 16];
axis(v);
k=0:40;
x=[k];
y=filter(num,den,x);
plot(k,y,'o',k,y,'-',k,k,'--')
grid
title('Respuesta a una rampa unitaria')
xlabel('k')
ylabel('y(k)')
```

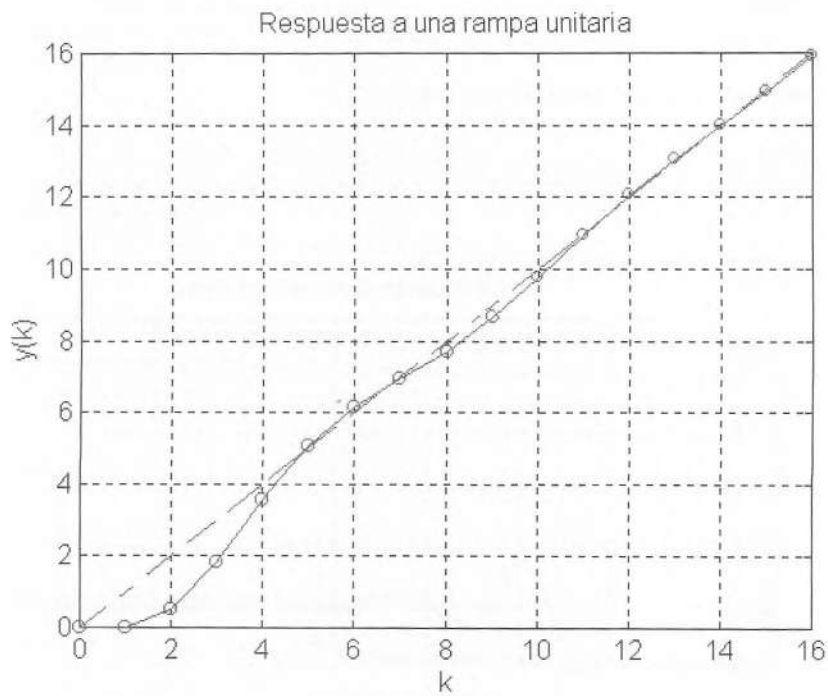


Figura 4-20

Los polos en lazo cerrado para el sistema se pueden obtener del polinomio característico, que es el denominador de la función de transferencia discreta en lazo cerrado.

$$\text{den}(z) = z^4 - 1.8528z^3 + 1.5906z^2 - 0.6642z + 0.0528$$

Introduciendo la sentencia

$$r = \text{roots}(\text{den})$$

obtenemos los polos en lazo cerrado para el sistema tal como sigue:

Programa en MATLAB 4-20

```
%-------Respuesta a una rampa unitaria-------

num=[0 0.5151 -0.1452 -0.2963 0.0528];
den=[1 -1.8528 1.5906 -0.6642 0.0528];
v=[0 16 0 16];
axis(v);
k=0:40;
x=[k];
y=filter(num,den,x);
plot(k,y,'o',k,y,'-',k,k,'--')
grid
title('Respuesta a una rampa unitaria')
xlabel('k')
ylabel('y(k)')
```

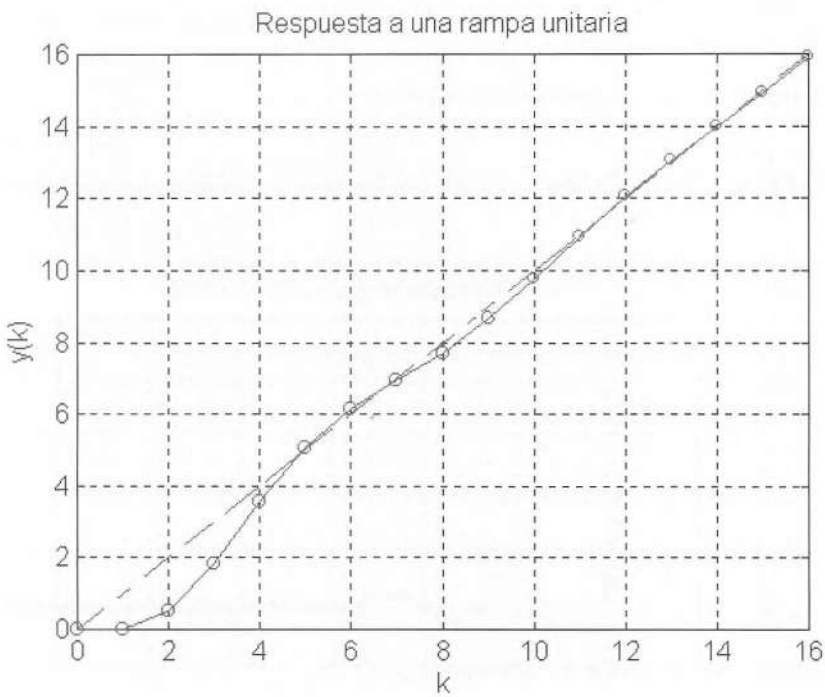


Figura 4-20

Los polos en lazo cerrado para el sistema se pueden obtener del polinomio característico, que es el denominador de la función de transferencia discreta en lazo cerrado.

$$\text{den}(z) = z^4 - 1.8528z^3 + 1.5906z^2 - 0.6642z + 0.0528$$

Introduciendo la sentencia

$$r = \text{roots}(\text{den})$$

obtenemos los polos en lazo cerrado para el sistema tal como sigue:

```
den =

    1.0000   -1.8528    1.5906   -0.6642    0.0528

r=roots(den)

r =

   0.4763+ 0.6521i
   0.4763- 0.6521i
   0.7989
   0.1013
```

Este sistema ha sido diseñado para tener error en estado estacionario nulo como respuesta a una entrada en rampa. Para verificar esto, calculamos el error en estado estacionario. La función de transferencia discreta para el sistema en lazo abierto es

$$G(z) = \frac{(1{,}4z^2 - 1{,}4z + 0{,}2)(0{,}3679z + 0{,}2642)}{(z-1)(z-0{,}3679)(z-1)z}$$

$$= \frac{0{,}5151(z-0{,}1727)(z-0{,}8273)(z+0{,}7181)}{z(z-0{,}3679)(z-1)^2}$$

La constante de error de velocidad K_v de este sistema es

$$K_v = \lim_{z\to 1}\frac{(1-z^{-1})G(z)}{T} = \lim_{z\to 1}\frac{(z-1)G(z)}{T}$$

$$= \lim_{z\to 1}\frac{0{,}5151(z-0{,}1727)(z-0{,}8273)(z+0{,}7181)}{Tz(z-0{,}3679)(z-1)} = \infty$$

por lo tanto, el error en estado estacionario para la respuesta a una entrada en rampa es:

$$\textit{error en estado estacionario} = \frac{1}{K_v} = 0$$

EJEMPLO 4-8: SISTEMA DE CONTROL DISEÑADO PARA ENTRADAS EN SALTO

Si la respuesta de un sistema de control en tiempo discreto a una entrada en salto presenta el mínimo tiempo de asentamiento posible (esto es, la salida alcanza el valor final en el tiempo mínimo y permanece allí), no existe error en estado estacionario y no se produce rizado entre los instantes de muestreo, entonces este tipo de respuesta se conoce normalmente como una *respuesta plana*.

Sea el sistema de control digital que se muestra en la Figura 4-21(a). La señal de error e(t), que es la diferencia entre la entrada r(t) y la salida c(t), se muestrea cada intervalo de tiempo de T seg. La entrada al controlador digital es la señal de error e(kT). La salida del controlador digital es la señal de control u(kT). La señal de control u(kT) se introduce en el retenedor de orden cero y la salida del retenedor, u(t), que es una función continua a tramos se lleva a la planta.

Un sistema de control en tiempo discreto se puede diseñar para conseguir el tiempo de asentamiento mínimo posible con error estacionario nulo como respuesta a una entrada en salto y además la salida no presenta rizado entre los instantes de muestreo una vez se ha alcanzado el estado estacionario.

Definamos la transformada z de la planta que viene precedida por el retenedor de orden cero como G(z), o

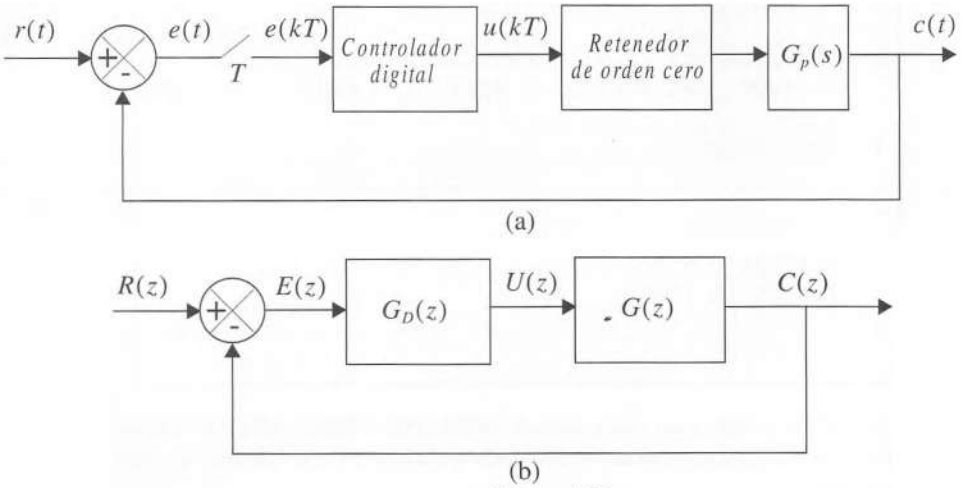


Figura 4-21

$$G(z) = Z\left[\frac{1-e^{-Ts}}{s}G_p(s)\right]$$

Entonces la función de transferencia discreta en lazo abierto es $G_D(z)G(z)$, tal como se muestra en la Figura 4-21(b).

Suponga que la función de transferencia de la planta $G_p(s)$ viene dada por:

$$G_p(s) = \frac{1}{s(s+1)}$$

Un controlador digital $G_D(z)$ se puede diseñar de forma tal que el sistema en lazo cerrado presentará una respuesta plana a una entrada tipo salto unitario. El periodo de muestreo T se supone que es 1 seg.

La transformada z de la planta que viene precedida por el retenedor de orden cero es:

$$G(z) = Z\left[\frac{1-e^{-Ts}}{s}\frac{1}{s(s+1)}\right]$$

$$= (1-z^{-1})Z\left[\frac{1}{s^2(s+1)}\right]$$

$$= (1-z^{-1})\left[\frac{z^{-1}}{(1-z^{-1})^2} - \frac{1}{1-z^{-1}} + \frac{1}{1-0{,}3679z^{-1}}\right]$$

$$= \frac{0{,}3679(1+0{,}7181z^{-1})z^{-1}}{(1-z^{-1})(1-0{,}3679z^{-1})}$$

Utilizando un método de diseño analítico, se puede diseñar el siguiente controlador de respuesta plana

$$G_D(z) = \frac{1{,}5820 - 0{,}5820z^{-1}}{1+0{,}4180z^{-1}}$$

(Para un estudio detallado de la técnica de diseño, véase la Referencia 3).

La función de transferencia discreta en lazo abierto para el sistema es:

$$G_D(z)G(z) = \frac{1{,}5820 - 0{,}5820z^{-1}}{1+0{,}4180z^{-1}}\frac{0{,}3679(1+0{,}7181z^{-1})z^{-1}}{(1-z^{-1})(1-0{,}3679z^{-1})}$$

$$= \frac{0{,}5820(z+0{,}7181)}{(z+0{,}4180)(z-1)}$$

La función de transferencia discreta en lazo cerrado para el sistema es:

$$\frac{C(z)}{R(z)} = \frac{0{,}5820(z+0{,}7181)}{0{,}5820(z+0{,}7181)+(z+0{,}4180)(z-1)} = \frac{0{,}5820z+0{,}4180}{z^2}$$

Representamos la respuesta a un salto unitario hasta k = 20. El Programa 4-21 en MATLAB dará el resultado deseado. La Figura 4-22 muestra la respuesta plana.

Es interesante obervar la respuesta a una entrada en rampa unitaria de este sistema. Este sistema está diseñado para exhibir respuesta plana para entradas tipo salto. Tales sistemas darán una "mejor respuesta" para entradas en salto, pero no será tan buena para otros tipos de entradas. (Para obtener la mejor respuesta para entradas de tipo rampa, el sistema debe diseñarse para dar respuesta plana con este tipo de entradas. Véase el Ejemplo 4-9).

Programa en MATLAB 4-21

```
%-------Respuesta a un escalón unitario-------

num=[0 0.5820 0.4180]; den=[1 0 0];
r=ones(1,21);
x=[1 ones(1,40)];
k=0:20;
v=[0 20 0 1.6]; axis(v);
c=filter(num,den,r);
plot(k,c,'o',k,c,'-')
grid, title('Respuesta a un escalón unitario de un sistema de respuesta plana')
xlabel('k'), ylabel('c(k)')
```

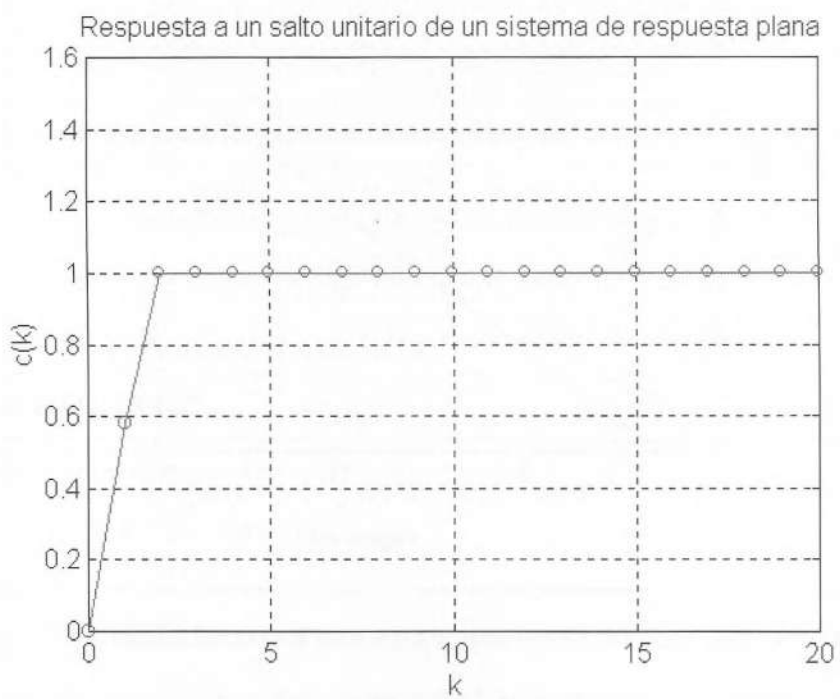


Figura 4-22

Para ver la respuesta a una entrada en rampa unitaria de este sistema, introduzca en el computador el Programa 4-22 en MATLAB. La Figura 4-23 muestra la respuesta a una entrada en rampa unitaria.

Programa en MATLAB 4-22

```
%-------Respuesta a una rampa unitaria-------

num=[0 0.5820 0.4180];
den=[1 0 0];
k=0:20;
r=[k];
v=[0 20 0 20];
axis(v);
c=filter(num,den,r);
plot(k,c,'o',k,c,'-',k,k,'--')
grid
title('Respuesta a una rampa unitaria de un sistema de respuesta plana')
xlabel('k')
ylabel('c(k)')
```

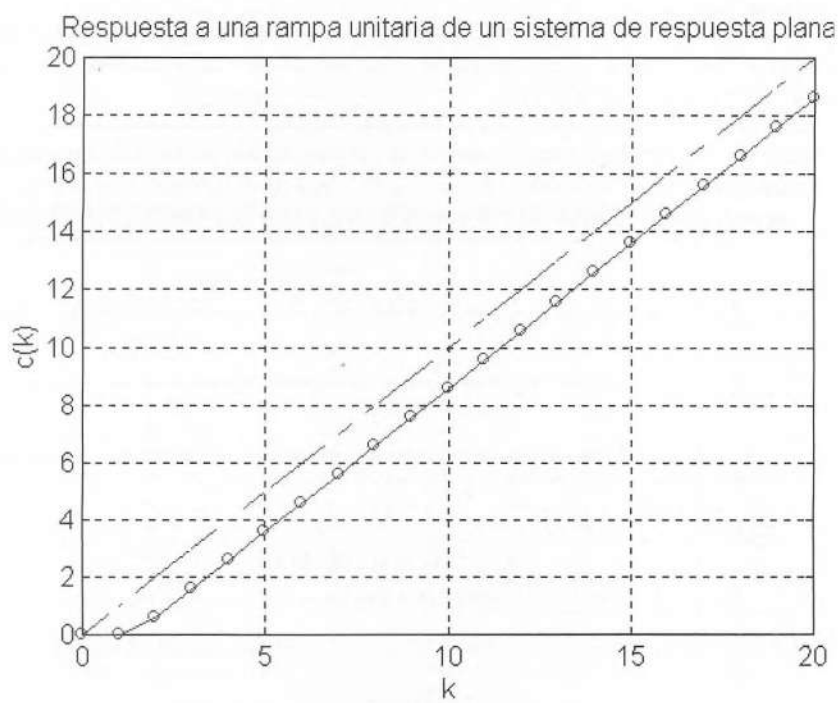


Figura 4-23

EJEMPLO 4-9: SISTEMA DE CONTROL PARA ENTRADAS EN RAMPA

Sea el sistema mostrado en la Figura 4-24. La función de transferencia discreta G(z) de la planta que está precedida por el retenedor de orden cero es:

$$G(z) = Z\left[\frac{1-e^{-Ts}}{s}\frac{1}{s^2}\right] = (1-z^{-1})Z\left[\frac{1}{s^3}\right]$$

$$= \frac{(1+z^{-1})z^{-}}{2(1-z^{-1})^{}}$$

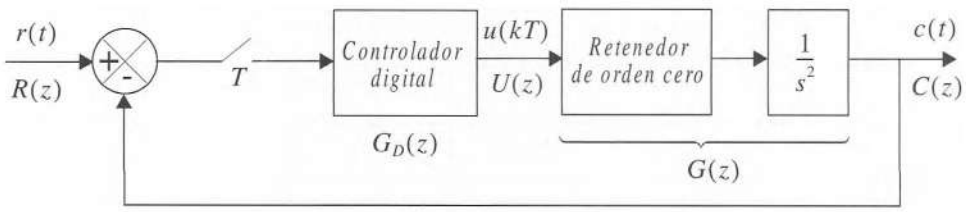


Figura 4-24

Se puede diseñar un controlador de respuesta plana de forma tal que presente tiempo de asentamiento mínimo con error en estado estacionario nulo para entradas tipo rampas y que no exhiba rizados entre muestras en el estado estacionario. El periodo de muestreo T se supone que es de 1 seg.

Este controlador se puede determinar empleando un método de diseño analítico. La función de transferencia discreta del controlador de respuesta plana para este sistema es

$$G_D(z) = \frac{2{,}5(1-0{,}6z^{-1})}{1+0{,}75z^{-1}}$$

(Para el diseño de este controlador de respuesta plana, véase la Referencia 3).

La función de transferencia discreta para el sistema en lazo abierto es

$$G_D(z)G(z) = \frac{2{,}5(1-0{,}6z^{-1})}{1+0{,}75z^{-1}}\,\frac{(1+z^{-1})\,z^{-1}}{2(1-z^{-1})^2}$$

$$= \frac{2{,}5(z-0{,}6)(z+1)}{(z+0{,}75)2(z-1)^2}$$

La función de transferencia discreta para el sistema en lazo cerrado es

$$\frac{C(z)}{R(z)} = \frac{2{,}5(z-0{,}6)(z+1)}{2{,}5(z-0{,}6)(z+1)+(z+0{,}75)2(z-1)^2}$$

$$= \frac{1{,}25z^2+0{,}5z-0{,}75}{z^3}$$

Programa en MATLAB 4-23

```
%-------Respuesta a una rampa unitaria-------

num=[0 1.25 0.5 -0.75]; den=[1 0 0 0];
k=0:20;
r=[k];
v=[0 20 0 20]; axis(v);
c=filter(num,den,r);
plot(k,c,'o',k,c,'-',k,k,'--')
grid, title('Respuesta a una rampa de un sistema de respuesta plana para
entradas tipo rampa')
xlabel('k'), ylabel('c(k)')
```

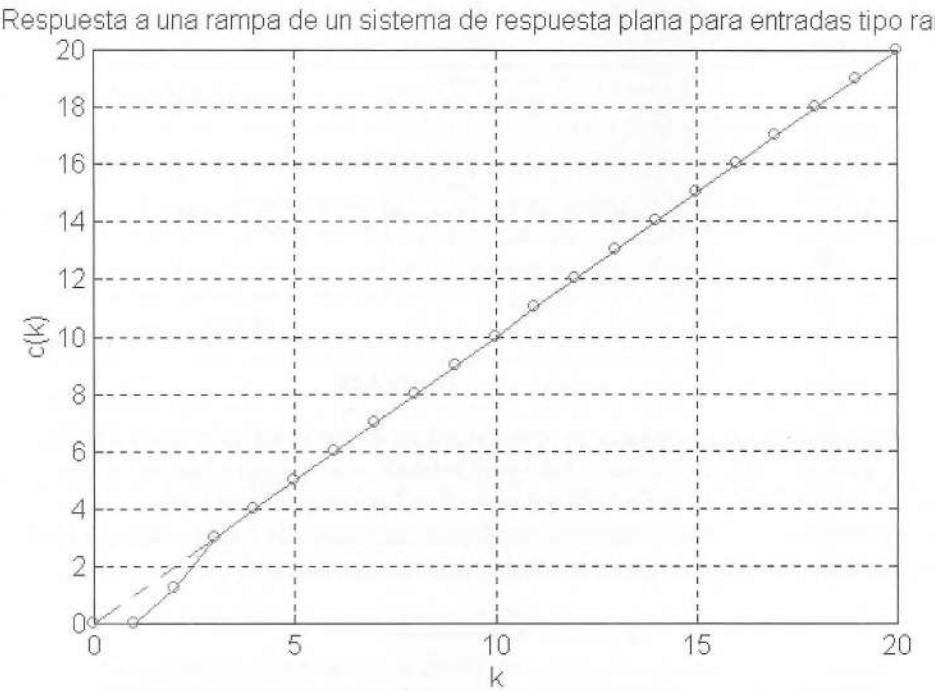


Figura 4-25

Se puede obtener una gráfica de la respuesta a una entrada en rampa unitaria introduciendo en el computador el Programa 4-23 en MATLAB. En la Figura 4-25 se muestra la gráfica resultante. Obsérvese que para $3 \leq k$ la salida sigue a la entrada en rampa sin error. Este sistema posee una respuesta "óptima" para la entrada en rampa, pero no sucede lo mismo con otros tipos de entradas. Por ejemplo, para una entrada en salto unitario la salida poseerá una sobreelongación antes de alcanzar el estado estacionario. El Programa 4-24 en MATLAB dará una gráfica de la respuesta a un salto unitario. En la Figura 4-26 se muestra esta gráfica

Programa en MATLAB 4-24

```
%-------Respuesta a un escalón unitario-------

num=[0 1.25 0.5 -0.75];
den=[1 0 0 0];
r=ones(1,21);
k=0:20;
v=[0 20 0 2];
axis(v);
c=filter(num,den,r);
plot(k,c,'o',k,c,'-')
grid
title('Respuesta a un escalón de un sistema de respuesta plana para entradas
tipo rampa')
xlabel('k')
ylabel('c(k)')
```

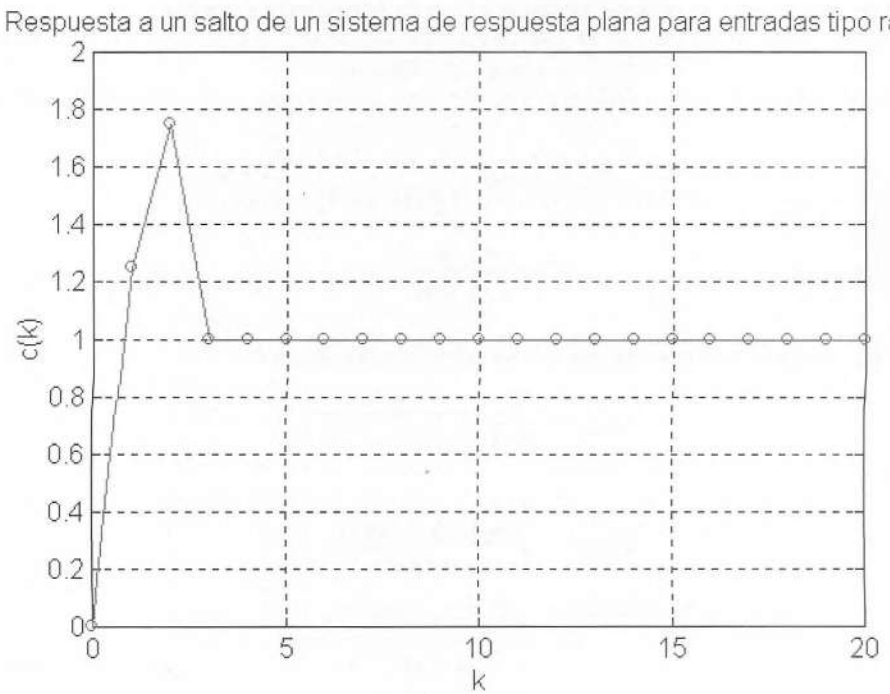


Figura 4-26

4.7 RESPUESTA TRANSITORIA DE SISTEMAS EN TIEMPO DISCRETO DEFINIDOS EN EL ESPACIO DE ESTADOS

En esta sección, presentaremos la respuesta transistoria de sistemas en tiempo discreto definidos en el espacio de estados. Específicamente, determinaremos la respuesta en salto del sistema

$$\mathbf{x}(k+1) = \mathbf{G}\mathbf{x}(k) + \mathbf{H}u(k)$$
$$y(k) = \mathbf{C}\mathbf{x}(k) + Du(k)$$

[La entrada $u(k)$ puede ser una entrada en salto, en rampa, una delta de Kronecker, etc.]
Como ejemplo consideremos el sistema siguiente:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0{,}16 & -1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) \tag{4-2}$$

$$y(k) = \begin{bmatrix} 1{,}16 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \tag{4-3}$$

Para este sistema

$$\mathbf{G} = \begin{bmatrix} 0 & 1 \\ -0{,}16 & -1 \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1{,}16 & 1 \end{bmatrix}, \quad D = 0$$

Supongamos que el sistema está inicialmente en reposo y que $u(k)$ es una secuencia de salto unitario.

En primer lugar, daremos una solución analítica a este problema. La función de transferencia discreta $Y(z)/U(z)$ de este sistema se obtiene mediante el uso de la siguiente ecuación:

$$\frac{Y(z)}{U(z)} = \mathbf{C}(z\mathbf{I} - \mathbf{G})^{-1}\mathbf{H} + D$$

$$= \begin{bmatrix} 1{,}16 & 1 \end{bmatrix} \begin{bmatrix} z & -1 \\ 0{,}16 & z+1 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + 0$$

$$= \frac{z + 1{,}16}{z^2 + z + 0{,}16}$$

Para una secuencia de entrada que sea un salto unitario, tenemos

$$U(z) = \frac{z}{z-1}$$

por tanto

$$Y(z) = \frac{z + 1{,}16}{z^2 + z + 0{,}16} \frac{z}{z-1}$$

$$= \frac{z^{-1} + 1{,}16z^{-2}}{1 + z^{-1} + 0{,}16z^{-2}} \frac{1}{1 - z^{-1}}$$

$$= \frac{z^{-1} + 1{,}16z^{-2}}{1 - 0{,}84z^{-2} - 0{,}16z^{-3}}$$

$$= z^{-1} + 1{,}16z^{-2} + 0{,}84z^{-3} + 1{,}1344z^{-4} + \ldots$$

Obsérvese que el valor final $y(\infty)$ se puede obtener mediante el teorema del valor final como sigue

$$y(\infty) = \lim_{z \to 1}(1 - z^{-1}) \frac{z^{-1} + 1{,}16z^{-2}}{1 + z^{-1} + 0{,}16z^{-2}} \frac{1}{1 - z^{-1}}$$

$$= \frac{2{,}16}{2{,}16} = 1{,}0000$$

Por tanto $y(k)$ se puede expresar como

$$y(0) = 0$$
$$y(1) = 1$$
$$y(2) = 1.16$$
$$y(3) = 0.84$$
$$y(4) = 1.1344$$
$$\vdots$$
$$y(\infty) = 1.0000$$

A continuación, determinaremos $x_2(k)$. Teniendo en cuenta (4-2), tenemos

$$x_1(k+1) = x_2(k)$$

$$x_2(k+1) = -0{,}16x_1(k) - x_2(k) + u(k)$$

Por tanto, obtenemos

$$x_2(k+2) + x_2(k+1) + 0{,}16x_2(k) = u(k+1)$$

Tomando la transformada z de esta ecuación, teniendo en cuenta que $x_2(0) = 0$, $x_2(1) = 1$ y $u(0) = 1$, encontramos:

$$z^2X_2(z) - z + zX_2(z) + 0{,}16X_2(z) = zU(z) - z$$

o

$$X_2(z) = \frac{z}{z^2 + z + 0{,}16}U(z)$$

$$= \frac{z^{-1}}{1 + z^{-1} + 0{,}16z^{-2}} \frac{1}{1 - z^{-1}}$$

$$= \frac{z^{-1}}{1 - 0{,}84z^{-2} - 0{,}16z^{-3}}$$

$$= z^{-1} + 0{,}84z^{-3} + 0{,}16z^{-4} + 0{,}7056z^{-5} + \ldots$$

El valor final de $x_2(k)$ es

$$x_2(\infty) = \lim_{z \to 1}(1 - z^{-1}) \frac{z^{-1}}{1 + z^{-1} + 0{,}16z^{-2}} \frac{1}{1 - z^{-1}}$$

$$= \frac{1}{2{,}16} = 0{,}4630$$

Por tanto,

$$\begin{aligned}
&x_2(0) = 0\\
&x_2(1) = 1\\
&x_2(2) = 0\\
&x_2(3) = 0.84\\
&x_2(4) = 0.16\\
&x_2(5) = 0.7056\\
&\quad\vdots\\
&x_2(\infty) = 0.4630
\end{aligned}$$

Para determinar $x_1(k)$ conviene observar que $x_1(0) = 0$ y $x_1(k+1) = x_2(k)$. Por tanto

$$\begin{aligned}
&x_2(0) = 0\\
&x_2(1) = x_2(0) = 1\\
&x_2(2) = x_2(1) = 0\\
&x_2(3) = x_2(2) = 0.84\\
&x_2(4) = x_2(3) = 0.16\\
&x_2(5) = x_2(4) = 0.7056\\
&\quad\vdots\\
&x_2(\infty) = x_2(\infty) = 0.4630
\end{aligned}$$

Solución con MATLAB

Para calcular la salida $y(k)$ con MATLAB, en primer lugar convertimos las ecuaciones expresadas en el espacio de estados (4-2) y (4-3) en la función de transferencia discreta $Y(z)/U(z)$ mediante el uso de la siguiente orden

```
[num,den] = ss2tf(G,H,C,D)
```

a continuación utilizamos la orden filter

```
y = filter(num,den,u)
```

donde u es la secuencia de salto unitario.

Para obtener la respuesta $x_1(k)$, utilice la siguiente ecuación de salida ficticia:

$$x_1(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \mathbf{F}\mathbf{x}(k)$$

donde

$$\mathbf{F} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Después convierta las ecuaciones en el espacio de estados (4-2) y (4-4) en la función de transferencia discreta $X_1(z)/U(z)$ con la orden

```
[num1,den1] = ss2tf(G,H,F,D)
```

y finalmente utilice otra vez la orden filter.

```
x1 = filter(num1, den1, u)
```

Análogamente, para obtener la respuesta $x_2(k)$, considere otra ecuación de salida ficticia:

$$x_2(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \mathbf{J}\mathbf{x}(k)$$

donde

$$\mathbf{J} = [0 \quad 1]$$

Convierta las ecuaciones en el espacio de estados (4-2) y (4-5) en la función de transferencia discreta $X_2(z)/U(z)$ con la orden

```
[num2,den2] = ss2tf(G,H,J,D)
```

y utilice la orden filter

```
x2 = filter(num2,den2,u)
```

Con el Programa 4-25 en MATLAB se obtienen las respuestas de y(k), x1(k) y x2(k) respecto de k. Las Figuras 4-27, 4-28 y 4-29 muestran respectivamente estas funciones.

Programa en MATLAB 4-25

```
%-------Respuesta a un escalón de sistemas en tiempo discreto definidos en el
%espacio de estados-------

%*****La respuesta a un escalón de sistemas en tiempo discreto definidos en el
%espacio de estados se obtendrá convirtiendo primero las ecuaciones en el
%espacio de estados en funcion de transferencia discreta y despues utilizando
%la orden 'filter' para obtener la respuesta a un escalón*****

%*****Introduzca las matrices G, H, C, F, J, D*****

G=[0 1;-0.16 -1]; H=[0;1];
C=[1.16 1]; F=[1 0];
J=[0 1]; D=[0];

%*****Para obtener y(k) convertir las ecuaciones del espacio de estados en
%función de transferencia discreta Y(z)/U(z)*****

[num,den]=ss2tf(G,H,C,D);

%*****Introducir la orden para obtener la respuesta a un escalón unitario*****

u=ones(1,51); axis([0 50 -0.5 2]);
k=0:50;
y=filter(num,den,u);
plot(k,y,'o',k,y,'-'), grid, title('Respuesta a un escalón unitario de y(k)')
xlabel('k'), ylabel('y(k)')

%*****Para obtener x1(k) convertir las ecuaciones del espacio de estados en
%funcion de transferencia discreta X1(z)/U(z)*****

[num1,den1]=ss2tf(G,H,F,D);

%*****Introduzca la orden para obtener la respuesta a un escalón unitario*****

axis([0 50 -0.5 1.5]);
x1=filter(num1,den1,u);
plot(k,x1,'o',k,x1,'-'), grid, title('Respuesta a un escalón unitario de
x1(k)')
xlabel('k'), ylabel('x1(k)')

%*****Para obtener x2(k) convertir las ecuaciones del espacio de estados en
%funcion de transferencia discreta X2(z)/U(z)*****

[num2,den2]=ss2tf(G,H,J,D);

%*****Introduzca la orden para obtener la respuesta a un escalón unitario*****

axis([0 50 -0.5 1.5]);
x2=filter(num2,den2,u);
plot(k,x2,'o',k,x2,'-'), grid, title('Respuesta a un escalón unitario de
x2(k)')
xlabel('k'), ylabel('x2(k)')
```

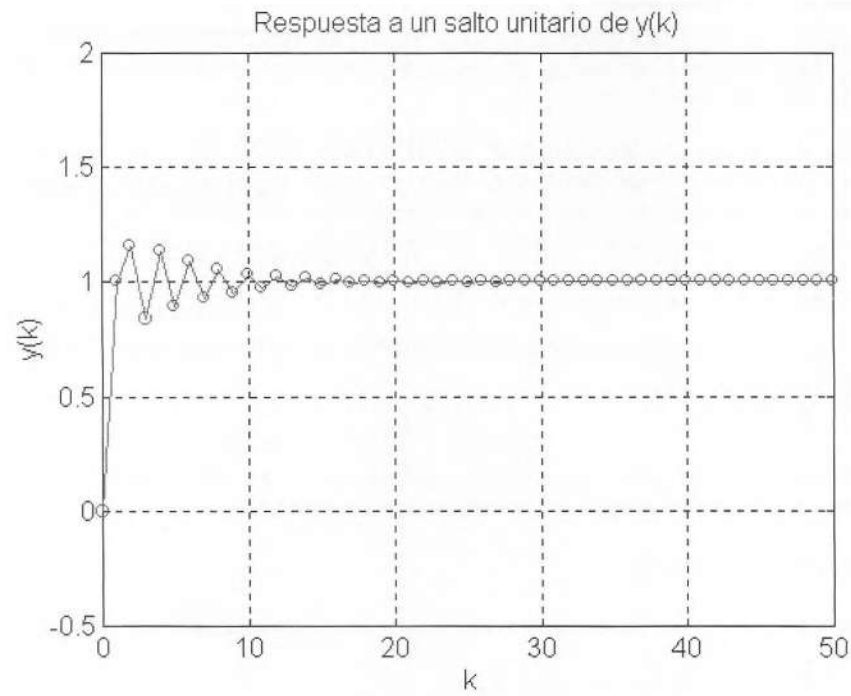


Figura 4-27

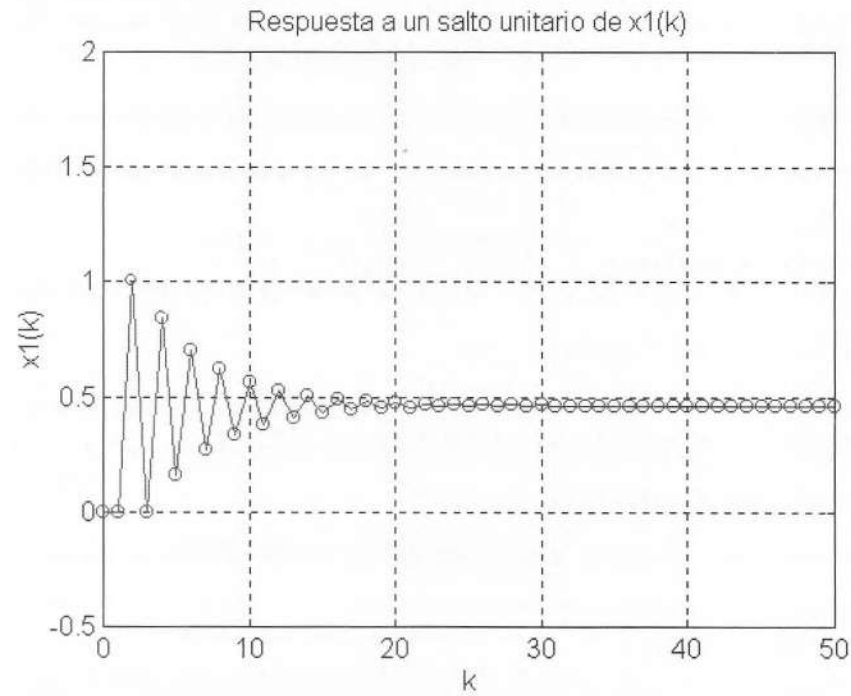


Figura 4-28

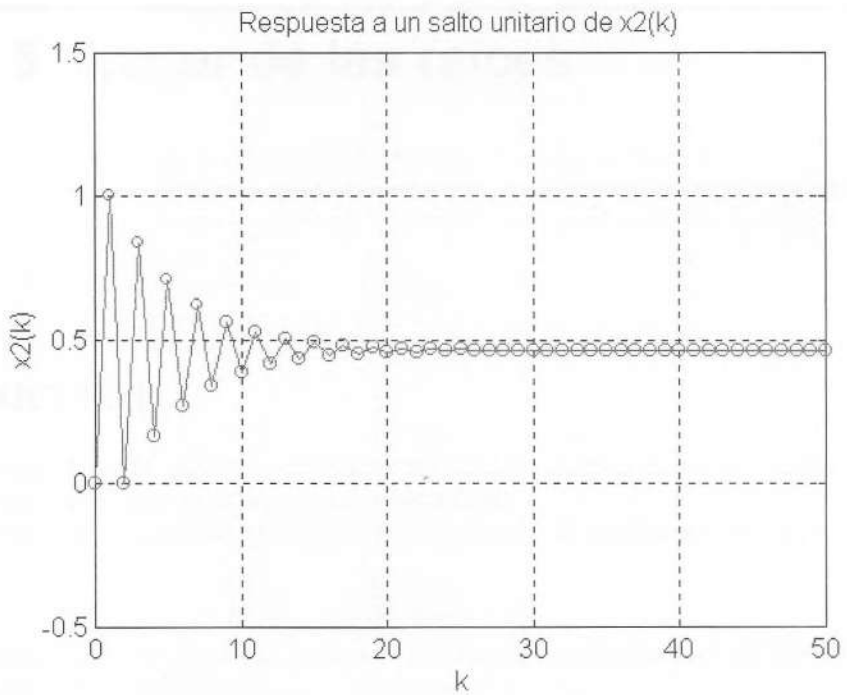

Respuesta a un salto unitario de x2(k)
1.5
1
0.5
0
-0.5
x2(k)
0
10
20
30
40
50
k

Figura 4-29

Capítulo 5 Lugar de las raíces

5.1 INTRODUCCIÓN

Este capítulo trata el estudio del dibujo de lugar de las raíces de sistemas de control. La orden rlocus genera el lugar de las raíces de un sistema de una entrada y una salida.

Considere el sistema mostrado en la Figura 5-1. La función de transferencia en lazo cerrado es

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)} \tag{5-1}$$

La ecuación característica para este sistema en lazo cerrado, se obtiene igualando el denominador del lado derecho de la Eq.(5-1) a cero. Es decir,

$$1 + G(s)H(s) = 0$$

o

$$G(s)H(s) = -1 \tag{5-2}$$

Aquí, asumimos que $G(s)H(s)$ es un producto de polinomios en s. Se tiene que $G(s)H(s)$ es una cantidad compleja, la Eq.(5-2) se puede dividir en dos ecuaciones igualando los ángulos y las magnitudes a ambos lados, respectivamente, para obtener

Condición de ángulo:

$$\angle G(s)H(s) = \pm 180°(2k + 1), \qquad k = 0, 1, 2, \tag{5-3}$$

Condición de magnitud:

$$|G(s)H(s)| = 1 \tag{5-4}$$

Los valores de s que satisfacen las condiciones de ángulo y magnitud son las raíces de la ecuación característica, o los polos del lazo cerrado. Una gráfica de los puntos del plano complejo que satisfacen la condición de ángulo únicamente es el lugar de las raíces. Las raíces de la ecuación característica (los polos del lazo cerrado) corresponden a un valor dado de ganancia que se puede determinar de la condición de magnitud.

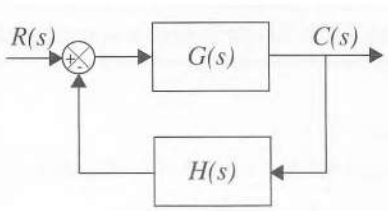


Figura 5-1

En muchos casos, $G(s)H(s)$ conlleva un parámetro ganancia K, y la ecuación característica se puede escribir como

$$1 + \frac{K(s+z_1)(s+z_2)\ldots(s+z_m)}{(s+p_1)(s+p_2)\ldots(s+p_n)} = 0 \tag{5-5}$$

Por tanto, el lugar de las raíces para el sistema es el lugar de los polos en lazo cerrado cuando la ganancia K varía de cero a infinito.

Para utilizar la orden *rlocus*, reescribimos la Eq.(5-5) como

$$1 + K\frac{\text{num}}{\text{den}} = 0$$

donde num es el polinomio numerador y den es el polinomio denominador.

Es decir,

$$\begin{aligned}\text{num} &= (s+z_1)(s+z_2)\ldots(s+z_m)\\ &= s^m + (z_1+z_2+\ldots+z_m)s^{m-1} + \ldots + z_1z_2\ldots z_m\\ \text{den} &= (s+p_1)(s+p_2)\ldots(s+p_n)\\ &= s^n + (p_1+p_2+\ldots+p_n)s^{n-1} + \ldots + p_1p_2\ldots p_n\end{aligned}$$

Obsérvese que los vectores num y den deben ser escritos en potencias descendentes de s.

Una orden de MATLAB con frecuencia utilizada para dibujar el lugar de las raíces es

```
rlocus(num,den)
```

Utilizando esta orden, la gráfica del lugar de las raíces se dibuja en la pantalla. El vector ganancia K se determina automáticamente. La orden rlocus funciona tanto para sistemas continuos como para discretos.

Para los sistemas definidos en el espacio de estados, rlocus(A,B,C,D) dibuja el lugar de las raíces del sistema y el vector ganancia se determina automáticamente.

Obsérvense esas órdenes

```
rlocus(num,den,K)    y    rlocus(A,B,C,D,K)
```

que utilizan un vector ganancia K suministrado por el usuario. (El vector K contiene todos los valores ganancia para los cuales los polos en lazo cerrado son calculados).

Si llamamos con los argumentos de mano izquierda

```
[r,K] = rlocus(num,den)
[r,K] = rlocus(num,den,K)
[r,K] = rlocus(A,B,C,D)
[r,K] = rlocus(A,B,C,D,K)
```

la pantalla mostrará la matriz r y el vector ganancia K (r tiene K filas y den - 1 columnas que contienen las localizaciones de las raíces complejas. Cada fila de la matriz corresponde a la ganancia del vector K). La orden plot

```
plot(r,' ')
```

dibuja el lugar de las raíces.

Si se desea representar el lugar de las raíces con marcas como 'o' o 'x', es necesario utilizar la siguiente orden:

```
r = rlocus(num,den)
plot(r,'o')    o    plot(r,'x')
```

Dibujar el lugar de las raíces utilizando marcas como 'o' o 'x' es instructivo, ya que cada polo en lazo cerrado calculado se muestra gráficamente; en alguna parte del lugar de las raíces esas marcas están situadas densamente y en otras partes estarán más separadas. MATLAB suministra su propio conjunto de valores de ganancia que utiliza para calcular el gráfico del lugar de las raíces. Esto lo hace mediante una rutina interna de paso variable. Además, MATLAB utiliza la característica de escalado automático de ejes de la orden *plot*.

Finalmente, obsérvese que al determinarse automáticamente el vector ganancia, los gráficos del lugar de las raíces de

$$G(s)H(s) = \frac{K(s+1)}{s(s+2)(s+3)}$$

$$G(s)H(s) = \frac{10K(s+1)}{s(s+2)(s+3)}$$

$$G(s)H(s) = \frac{200K(s+1)}{s(s+2)(s+3)}$$

son todos iguales. El num y el den es el mismo para los tres sistemas. El num y el den son

```
num = [0  0  1  1]
den = [1  5  6  0]
```

Estructura del capítulo

La sección 5-1 presenta el material introductorio para las órdenes del lugar de las raíces y los temas asociados. La sección 5-2 explica con detalle el dibujo del lugar de las raíces de varios sistemas. Un gran número de muestras de lugares de las raíces se muestran en esta sección. La sección 5-3 trata los casos especiales, donde los lugares de las raíces muestran algunas figuras asimétricas. La sección 5-4 presenta métodos para dibujar dos (o más) lugares de las raíces (como puede ser el lugar de las raíces del sistema y sus asíntotas) en una sola gráfica. Finalmente, la sección 5-5 muestra el dibujo del lugar de las raíces en el plano z (para sistemas discretos).

5.2 OBTENCIÓN DEL LUGAR DE LAS RAÍCES

En esta sección, se discutirá con detalle cómo obtener el lugar de las raíces utilizando varios sistemas de ejemplo. Se presentan un gran número de típicos lugares de las raíces con sus pragramas en MATLAB.

EJEMPLO 5-1

Considere el sistema de control mostrado en la Figura 5-2. La función de transferencia en lazo abierto es

$$G(s) = \frac{K(s^2+1)}{s(s+2)} = \frac{K(s^2+1)}{s^2+2s}$$

El sistema tiene los ceros en lazo abierto en $s = j$ y $s = -j$. Los polos en lazo abierto están en $s = 0$ y $s = -2$. Para obtener el lugar de las raíces para este sistema, se debe de introducir el programa de MATLAB 5-1 en el computador.

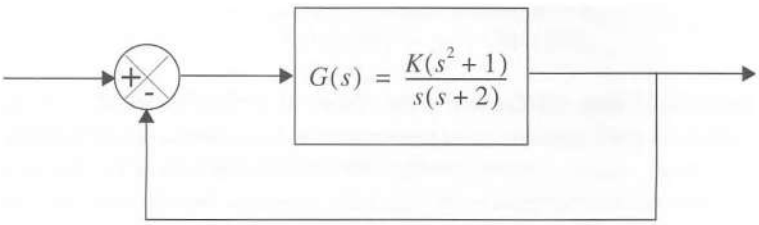


Figura 5-2

Al escribir el vector del numerador (num), los coeficientes del numerador no se deben de multiplicar por la ganancia K. Es decir, en lugar de

num = [K 0 K]

num debe de ser

num = [1 0 1]

El lugar de las raíces que se obtiene de la salida del computador se muestra en la Figura 5-3.

Programa en MATLAB 5-1

```
%-------Lugar de las raíces-------

num=[1 0 1];
den=[1 2 0];

rlocus(num,den)
grid
title('Lugar de las raíces de G(s)=K(s^2+1)/[s(s+2)]')
```

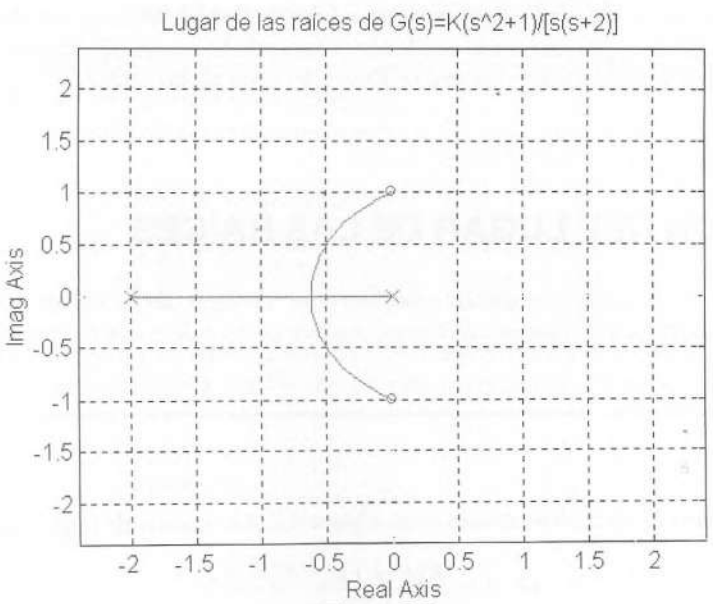


Figura 5-3

EJEMPLO 5-2

Considere el sistema de control mostrado en la Figura 5-4. La función de transferencia en lazo abierto es

$$G(s) = \frac{K(s+2)}{s^2+2s+3}$$

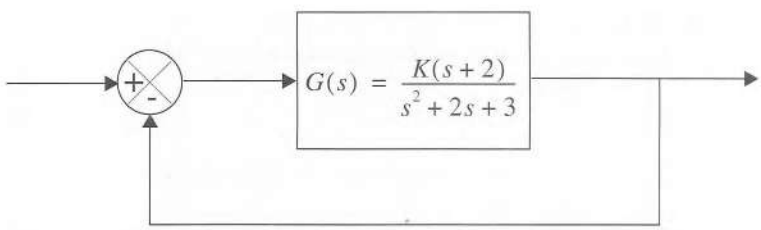


Figura 5-4

Los ceros y polos en lazo abierto están localizados en:

ceros del lazo abierto: $s = -2$

polos del lazo abierto: $s = -1 + j\sqrt{2}$, $s = -1 - j\sqrt{2}$

Un programa de MATLAB para obtener el lugar de las raíces pra este sistema se muestra en el programa de MATLAB 5-2. El lugar de las raíces generado por el computador se muestra en la Figura 5-5. Obsérvese que el lugar de las raíces se sitúa en el extremo izquierdo del diagrama. Si se desea dibujar el lugar de las raíces cerca del centro del diagrama, es necesario anular la característica de escalado automático de ejes de la orden plot y seleccionar manualmente los límites del dibujo. Si se desea seleccionar el eje x desde -5 hasta 1 y el eje y desde -2.5 hasta 2.5, introduzca la siguiente orden:

```
v = [-5  1  -2.5  2.5]
axis(v)
```

Programa en MATLAB 5-2

```
%-------Lugar de las raíces-------

num=[0 1 2];
den=[1 2 3];

rlocus(num,den)
grid
title('Lugar de las raíces de G(s)=K(s+2)/(s^2+2s+3)')
```

El lugar de las raíces generado por el computador basado en el programa de MATLAB 5-3 se muestra en la Figura 5-6.

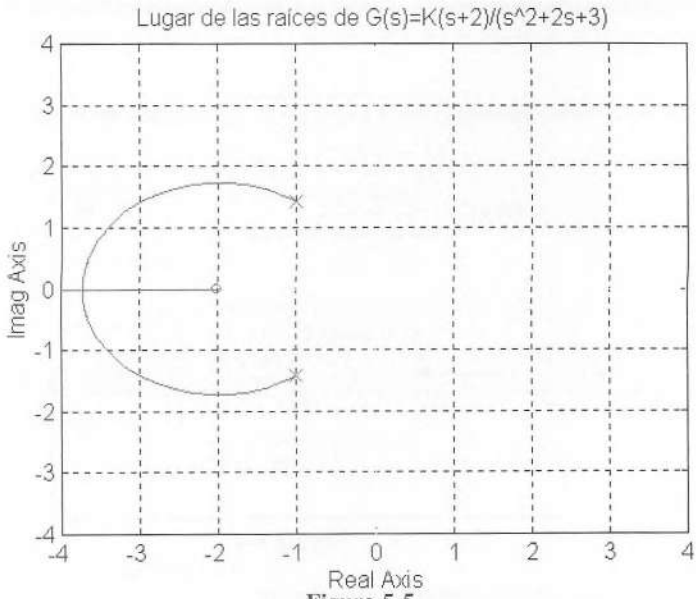


Figura 5-5

Programa en MATLAB 5-3

```
%-------Lugar de las raíces-------

num=[0 1 2];
den=[1 2 3];
rlocus(num,den)
v=[-5 1 -2.5 2.5];
axis(v);
grid
title('Lugar de las raíces de G(s)=K(s+2)/(s^2+2s+3)')
```

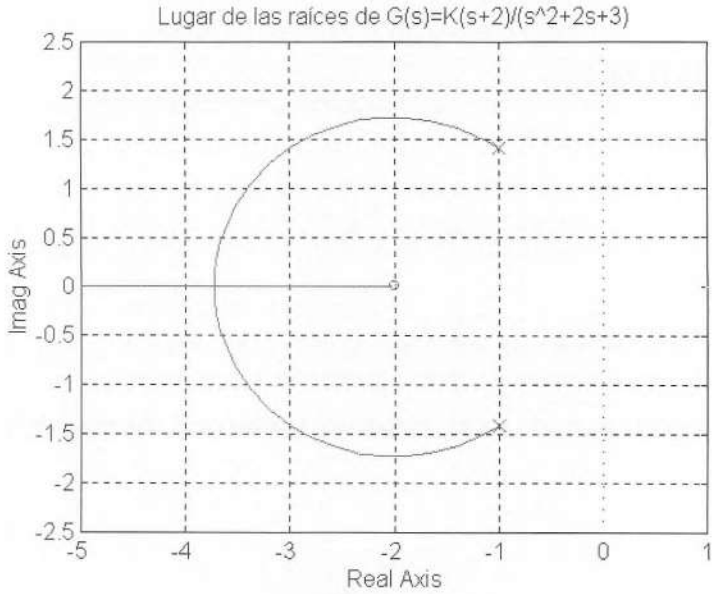


Figura 5-6

EJEMPLO 5-3

Considere el sistema de control mostrado en la Figura 5-7. Obsérvese que la variable ajustable k no aparece como un factor multiplicativo. La ecuación característica para el sistema es

$$+\frac{20(1+ks)}{s(s+1)(s+4)} = 0 \tag{5-6}$$

o

$$s^3 + 5s^2 + 4s + 20 + 20ks = 0 \tag{5-7}$$

Por definición

$$20k = K$$

y dividiendo ambos lados de la ecuación Eq.(5-7) por la suma de los términos que no contienen k, obtenemos

$$1 + \frac{Ks}{s^3 + 5s^2 + 4s + 20} = 0$$

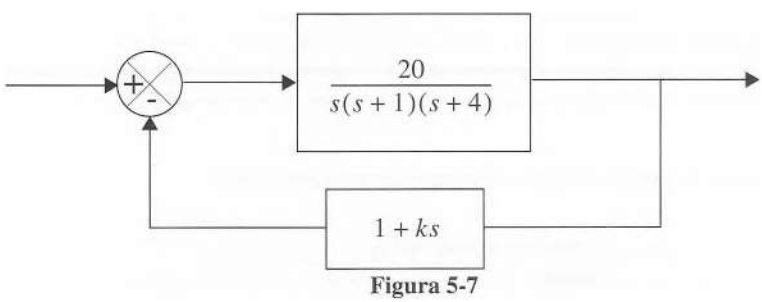


Figura 5-7

$$1 + \frac{Ks}{(s+j2)(s-j2)(s+5)} = 0 \tag{5-8}$$

La Eq.(5-8) la tenemos de la forma de la Eq.(5-5). El num y el den de este sistema son

num = [0 0 1 0]
den = [1 5 4 20]

Un diagrama de lugar de las raíces de este sistema se puede obtener introduciendo el programa de MATLAB 5-4 en el computador. El lugar de las raíces resultante se muestra en la Figura 5-8.

Programa en MATLAB 5-4

```
%-------Lugar de las raíces-------

num=[0 0 1 0];
den=[1 5 4 20];

rlocus(num,den)
grid
title('Lugar de las raíces de G(s)=Ks/[(s+5)(s^2+4)
```

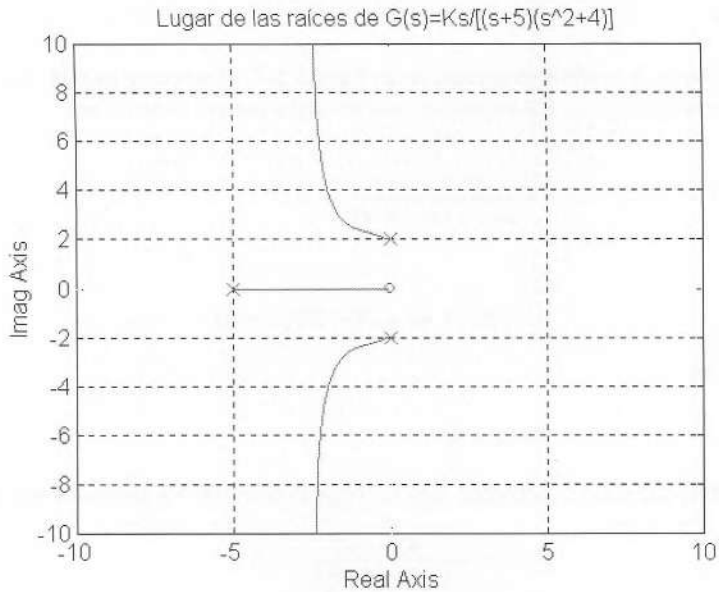


Figura 5-8

EJEMPLO 5-4

Un lugar de las raíces se puede dibujar utilizando la siguiente orden:

```
r = rlocus(num,den)
plot(r,' ')
```

Si utilizamos esta orden, los ceros y polos del lazo abierto no se mostrarán en el gráfico. Es necesario añadir pequeños círculos (o) y pequeñas cruces (x) a los ceros y polos en lazo abierto, respectivamente. Esto se puede realizar manualmente de una forma fácil.

Considere el sistema mostrado en la Figura 5-9, donde

$$G(s) = \frac{K(s+1)}{s^2(s+3{,}6)}, \qquad H(s) = 1$$

El programa de MATLAB 5-5 generará el lugar de las raíces para este sistema mostrado en la Figura 5-10. Obsérvese que los ceros y polos en lazo abierto no se muestran en el dibujo.

La Figura 5-11 muestra el mismo lugar de las raíces que el mostrado en la Figura 5-10, pero con los ceros y polos en lazo abierto marcados por 'o' y 'x', respectivamente. (Obsérvese que 'o' y 'x' son dibujados manualmente en el gráfico).

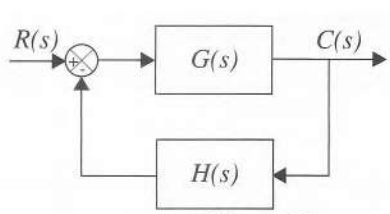


Figura 5-9

Programa en MATLAB 5-5

```
%-------Lugar de las raíces-------

num=[0 0 1 1];
den=[1 3.6 0 0];
v=[-6 4 -6 6];
axis(v);
r=rlocus(num,den);
plot(r,'-')
grid
title('Lugar de las raíces de G(s)=K(s+1)/[(s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imaginario')

%*****Observe que si representamos el lugar de las raíces utilizando la orden
%'r=rlocus(num,den),plot(r,'-')', entonces no se muestran en la gráfica los
%ceros y los polos en lazo abierto. Es necesario añadir pequeños círculos (o) y
%cruces (x)en los ceros y polos en lazo abierto resspectivamente.
%Esto se puede hacer fácilmente de forma manual.*****
```

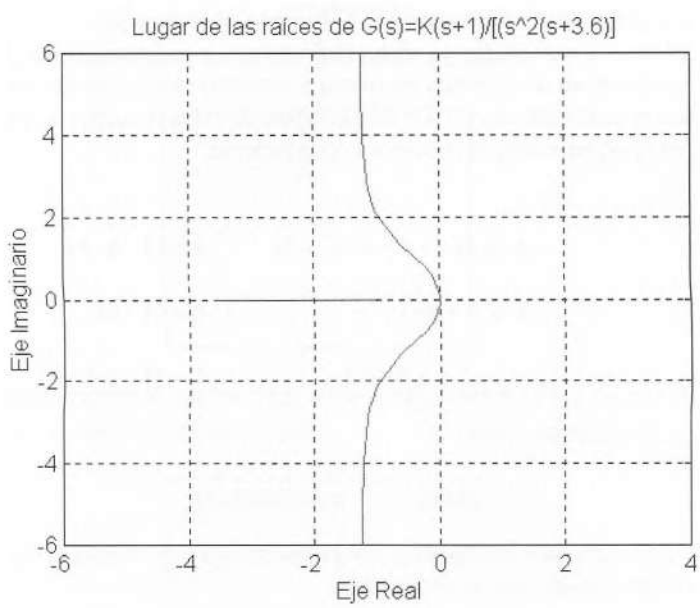


Figura 5-10

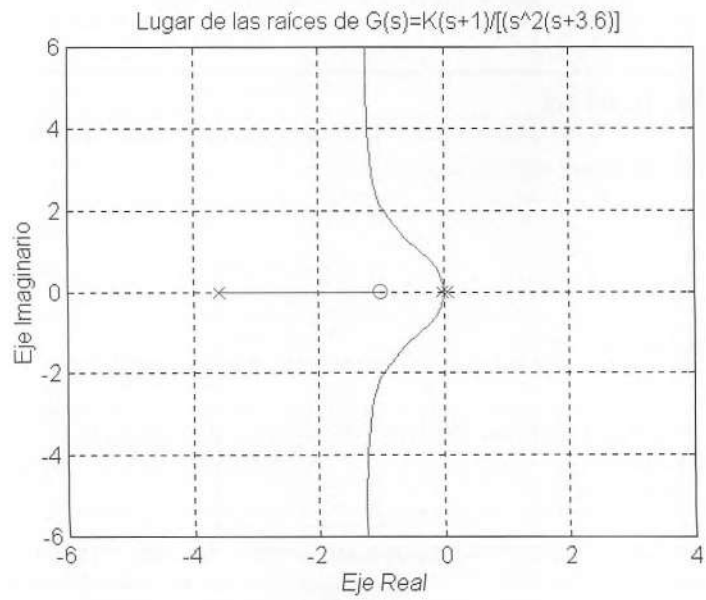


Figura 5-11

EJEMPLO 5-5

Considere el sistema de control mostrado en la Figura 5-12. Par representar el lugar de las raíces con MATLAB, es necesario el polinomio numerador y el polinomio denominador.

Para este problema, el numerador ya viene dado como un polinomio en s. Sin embargo, el denominador viene dado como un producto de términos de primer y segundo orden, con el resultado de multiplicar estos dos términos obtenemos el polinomio en s. La multiplicación de estos términos se puede realizar con facilidad utilizando la orden *convolution*, como se muestra a continuación.

Defina

$$a = s(s+4) = s^2 + 4s \quad : \quad \text{a} = [1 \quad 4 \quad 0]$$

$$b = s + 6 \quad : \quad \text{b} = [1 \quad 6]$$

$$c = s^2 + 1{,}4s + 1 \quad : \quad \text{c} = [1 \quad 1.4 \quad 1]$$

Después utilice la siguiente orden:

```
d = conv(a,b);    e = conv(c,d)
```

[Observe que conv(a,b) da el producto de los polinomios a y b]. Véase la siguiente salida del computador: El polinomio denominador que se obtiene es

```
den = [1   11.4   39   43.6   24   0]
```

```
a=[1 4 0];
b=[1 6];
c=[1 1.4 1];
d=conv(a,b)

d =

     1    10    24     0

e=conv(c,d)

e =

    1.0000   11.4000   39.0000   43.6000   24.0000         0
```

Para encontrar los ceros del lazo abierto de la función de transferencia dada, debemos utilizar la orden *roots*:

```
p = [1  2  4]
r = roots(p)
```

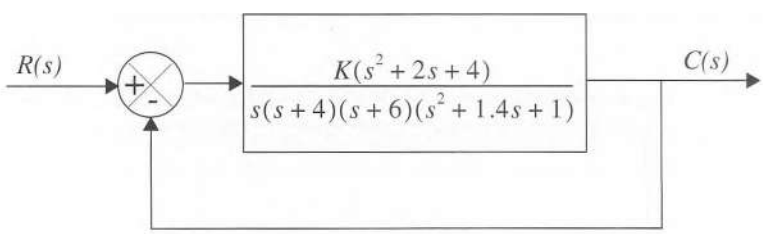


Figura 5-12

La orden y la salida del computador se muestran a continuación.

```
p=[1 2 4];
r=roots(p)

r =

  -1.0000 + 1.7321i
  -1.0000 - 1.7321i
```

Análogamente, para encontrar los polos complejos conjugados del lazo abierto (las raíces de $s^2 + 1.4s + 1 = 0$), debemos introducir la orden *roots* como sigue:

```
q=roots(c)

q =

  -0.7000 + 0.7141i
  -0.7000 - 0.7141i
```

Así, el sistema tiene los siguientes ceros y polos en lazo abierto:

ceros del lazo abierto: $s = -1 + j1{,}7321$, $s = -1 - j1{,}7321$

polos del lazo abierto: $s = -0{,}7 + j0{,}7141$, $s = -0{,}7 - j0{,}7141$

$s = 0$, $s = -4$, $s = -6$

El programa de MATLAB 5-6 dibujará el diagrama del lugar de las raíces para este sistema. El gráfico se muestra en la Figura 5-13.

Programa en MATLAB 5-6

```
%-------Lugar de las raíces-------

num=[0 0 0 1 2 4];
den=[1 11.4 39 43.6 24 0];
v=[-10 10 -10 10];
axis(v);
rlocus(num,den);

grid
title('Lugar de las raíces de G(s)=K(s^2+2s+4)/[s(s+4)(s+6)(s^2+1.4s+1)]')
```

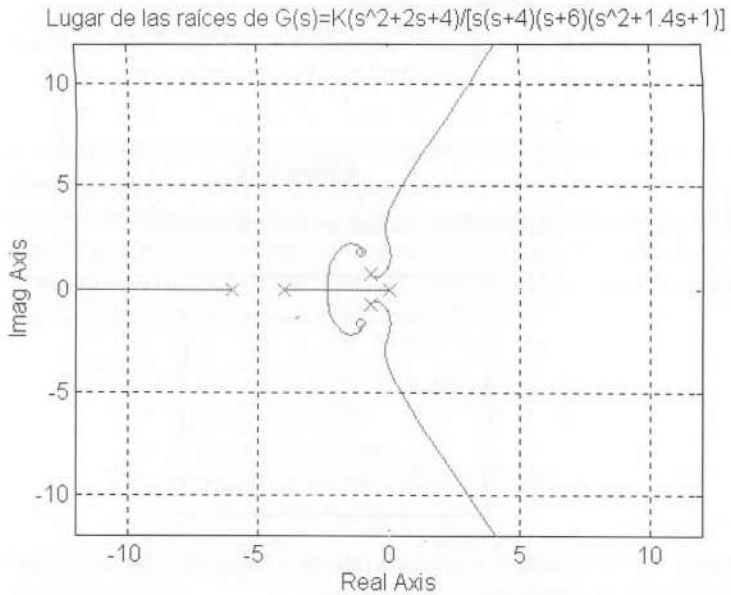


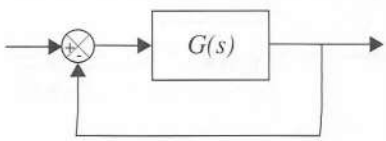
Figura 5-13

Ejemplos de lugares de las raíces de sistemas típicos en lazo cerrado

A continuación, se presentarán lugares de las raíces de varios sistemas típicos en lazo cerrado. Para cada sistema en lazo cerrado considerado se da un programa en MATLAB. También se señalan los ceros y polos en lazo abierto. Los diagramas de los lugares de las raíces de estos sistemas se muestran en los Ejemplos que van del 5-6 al 5-19.

EJEMPLO 5-6

Sistema

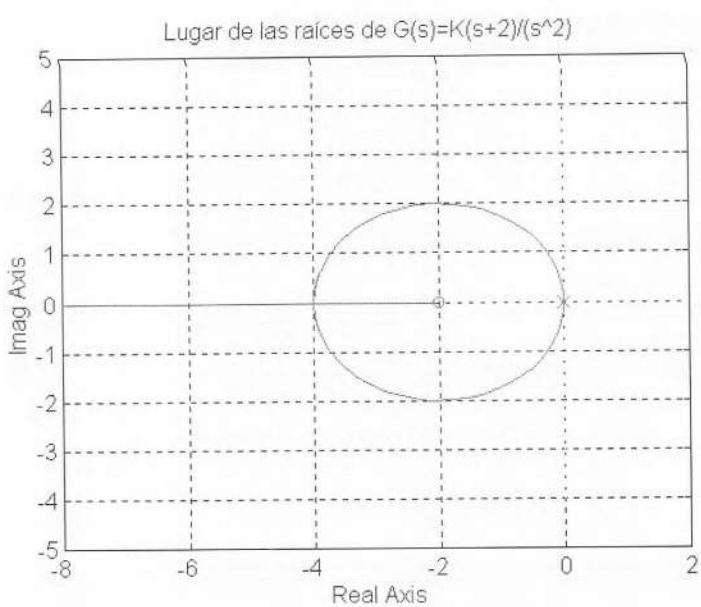


$$G(s) = \frac{K(s+2)}{s^2}$$

Programa en MATLAB

```
num=[0 1 2];
den=[1 0 0];
v=[-8 2 -5 5];
axis(v);
rlocus(num,den)

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(s)=K(s+2)/(s^2)')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -2$

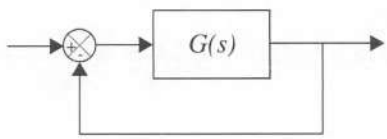
Polos en lazo abierto

$s = 0$

$s = 0$

EJEMPLO 5-7

Sistema

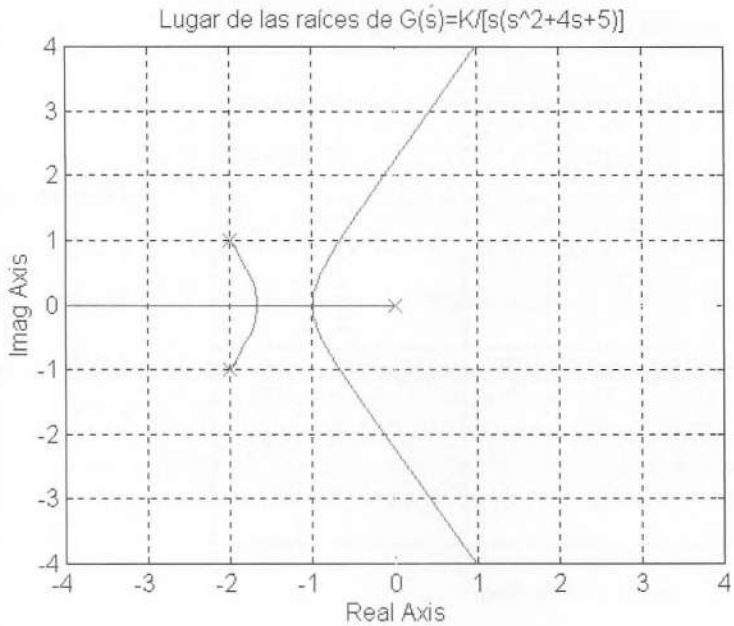


$$G(s) = \frac{K}{s(s^2 + 4s + 5)}$$

Programa en MATLAB

```
num=[0 0 0 1];
den=[1 4 5 0];
v=[-4 2 -4 4];
axis(v);
rlocus(num,den)

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(s)=K/[s(s^2+4s+5)]')
```

Lugar de las raíces



Ceros en lazo abierto

ninguno

Polos en lazo abierto

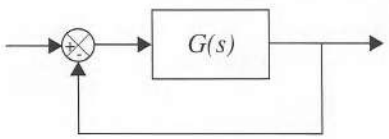
$s = -2 + j$

$s = -2 - j$

$s = 0$

EJEMPLO 5-8

Sistema

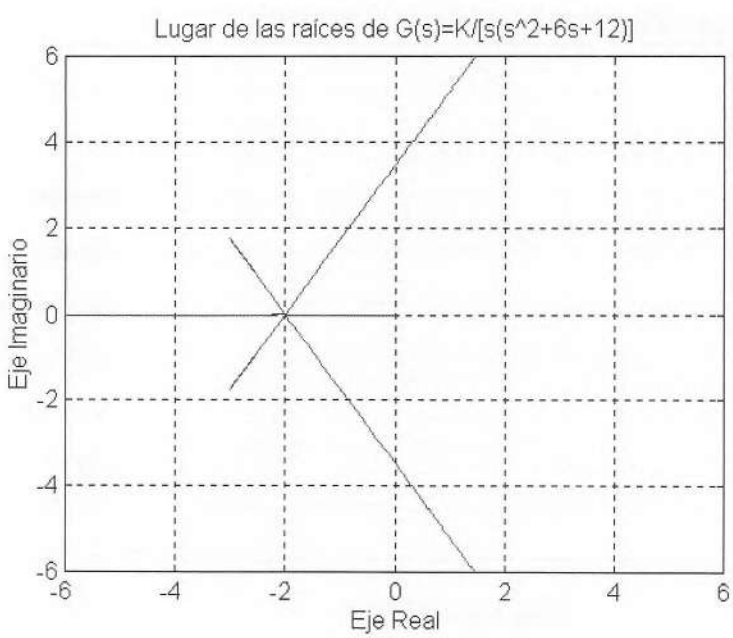


$$G(s) = \frac{K}{s(s^2 + 6s + 12)}$$

Programa en MATLAB

```
num=[0 0 0 1];
den=[1 6 12 0];
v=[-6 6 -6 6];
axis(v);
K1=0:0.1:10;
K2=10:5:400;
K=[K1 K2];
r=rlocus(num,den,K);
plot(r,'-')
grid
title('Lugar de las raíces de G(s)=K/[s(s^2+6s+12)]')
xlabel('Eje Real')
ylabel('Eje Imaginario')

%*****Observe que si representamos el lugar de las raíces utilizando la
%orden 'r=rlocus(num,den),plot(r,'-')', entonces no se muestran en la
%gráfica los ceros y los polos en lazo abierto. Es necesario añadir
%pequeños círculos (o) y cruces (x) en los ceros y polos en lazo abierto
%respectivamente. Esto se puede hacer fácilmente de forma manual.*****
```

Lugar de las raíces



Ceros en lazo abierto

ninguno

Polos en lazo abierto

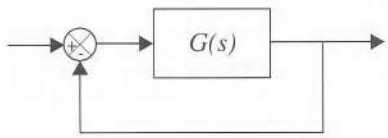
s = -3 + 1.7321j

s = -3 - 1.7321j

s= 0

EJEMPLO 5-9

Sistema

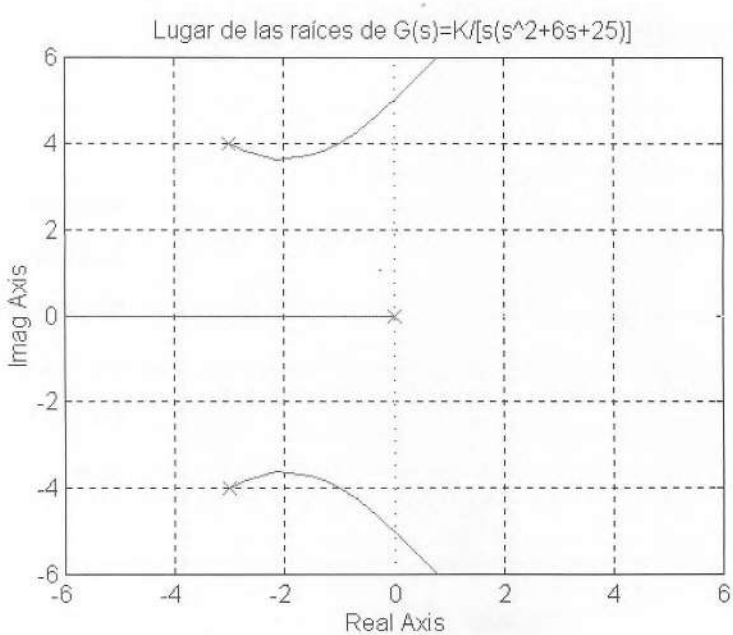


$$G(s) = \frac{K}{s(s^2 + 6s + 25)}$$

Programa en MATLAB

```
num=[0 0 0 1];
den=[1 6 25 0];
v=[-6 6 -6 6];
axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(s)=K/[s(s^2+6s+25)]')
```

Lugar de las raíces



Ceros en lazo abierto

ninguno

Polos en lazo abierto

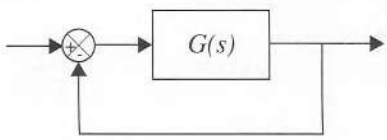
$s = -3 + 4j$

$s = -3 - 4j$

$s = 0$

EJEMPLO 5-10

Sistema

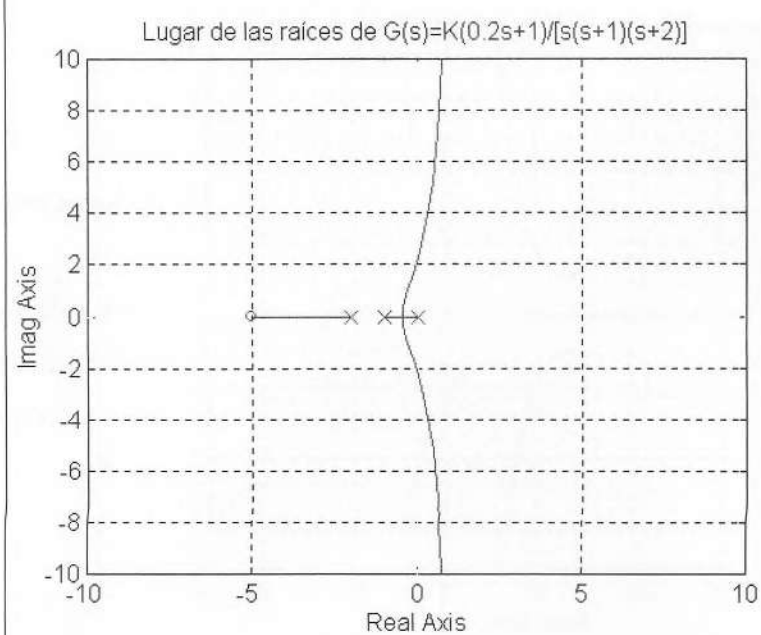


$$G(s) = \frac{K(0.2s+1)}{s(s+1)(s+2)}$$

Programa en MATLAB

```
num=[0 0 0.2 1];
den=[1 3 2 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K(0.2s+1)/[s(s+1)(s+2)]')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -5$

Polos en lazo abierto

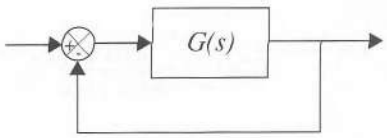
$s = 0$

$s = -1$

$s = -2$

EJEMPLO 5-11

Sistema

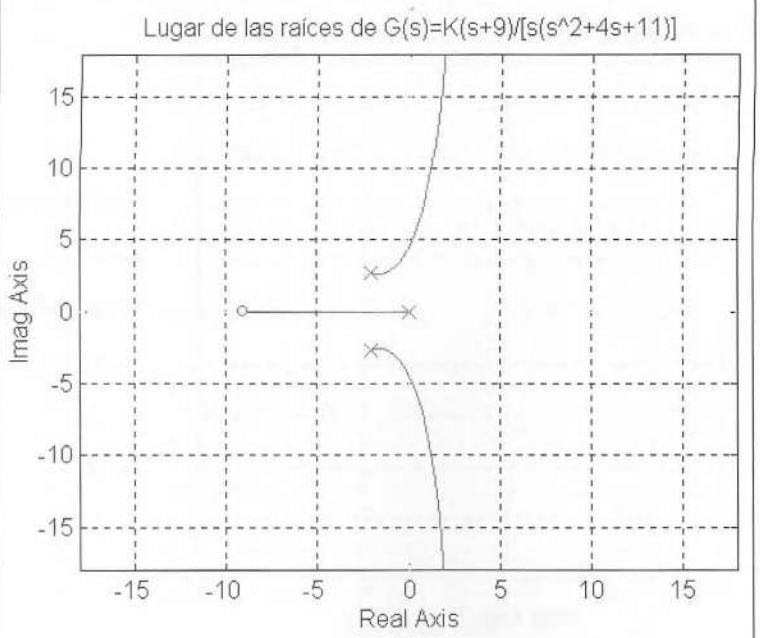


$$G(s) = \frac{K(s+9)}{s(s^2+4s+11)}$$

Programa en MATLAB

```
num=[0 0 1 9];
den=[1 4 11 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K(s+9)/[s(s^2+4s+11)]')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -9$

Polos en lazo abierto

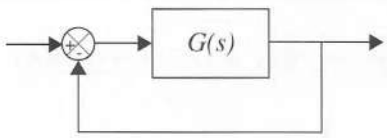
$s = -2 + 2.6458j$

$s = -2 - 2.6458j$

$s = 0$

EJEMPLO 5-12

Sistema

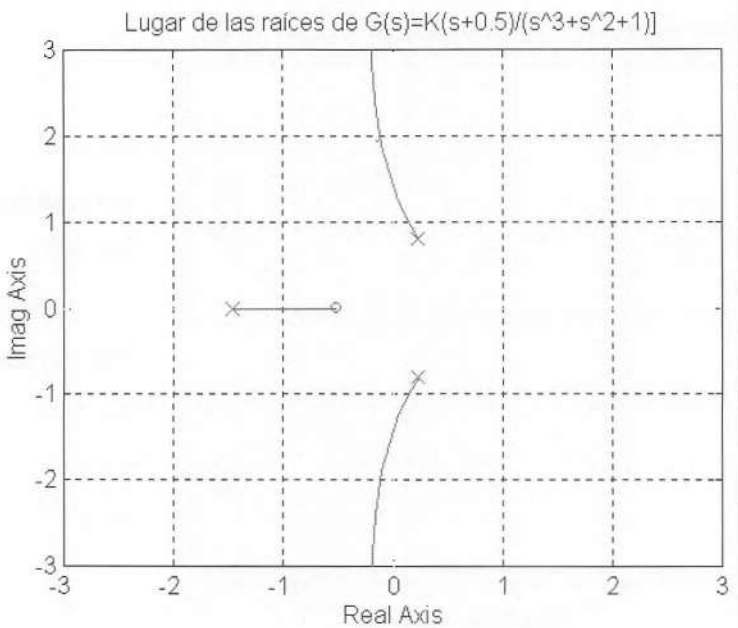


$$G(s) = \frac{K(s+0.5)}{s^3 + s^2 + 1}$$

Programa en MATLAB

```
num=[0 0 1 0.5];
den=[1 1 0 1];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K(s+0.5)/(s^3+s^2+1)]')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -0.5$

Polos en lazo abierto

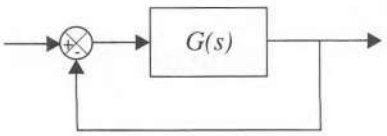
$s = 0.2328 + 0.7976j$

$s = 0.2328 - 0.7976j$

$s = -1.4656$

EJEMPLO 5-13

Sistema

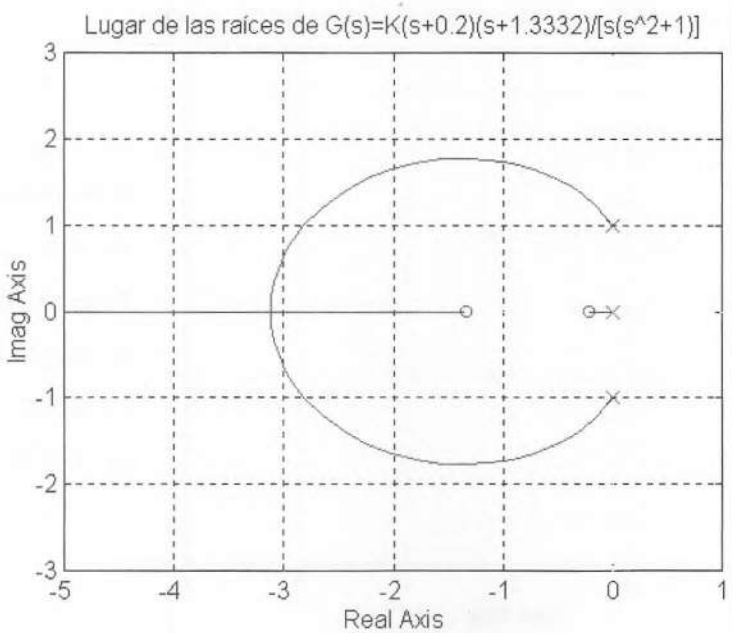


$$G(s) = \frac{K(s^2 + 1.5332s + 0.2666)}{s(s^2 + 1)}$$

Programa en MATLAB

```
num=[0 1 1.5332 0.2666];
den=[1 0 1 0];
v=[-5 1 -3 3];
axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(s)=K(s+0.2)(s+1.3332)/[s(s^2+1)]')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -0.2$

$s = -1.3332$

Polos en lazo abierto

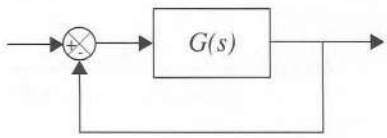
$s = 0$

$s = j$

$s = -j$

EJEMPLO 5-14

Sistema

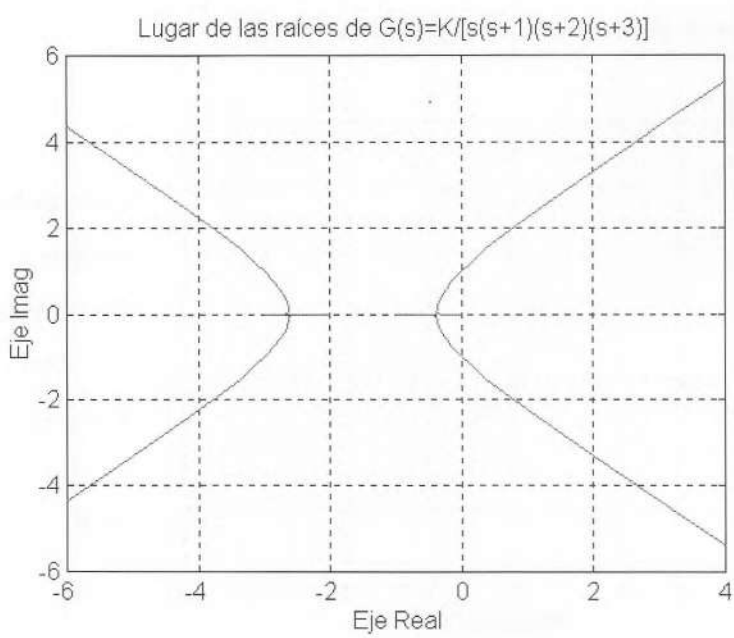


$$G(s) = \frac{K}{s(s+1)(s+2)(s+3)}$$

Programa en MATLAB

```
num=[0 0 0 0 1];
den=[1 6 11 6 0];
v=[-6 4 -6 6];
axis(v);
r=rlocus(num,den);

%*****Si desea dibujar la gráfica en color azul introduzca la siguiente
%orden*****

plot(r,'-b')

%*****Añadir rejilla, título y etiquetas en los ejes x e y*****

grid
title('Lugar de las raíces de G(s)=K/[s(s+1)(s+2)(s+3)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

Lugar de las raíces



Ceros en lazo abierto

ninguno

Polos en lazo abierto

$s = 0$

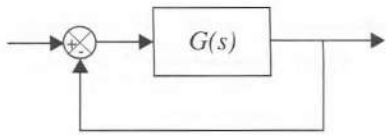
$s = -1$

$s = -2$

$s = -3$

EJEMPLO 5-15

Sistema

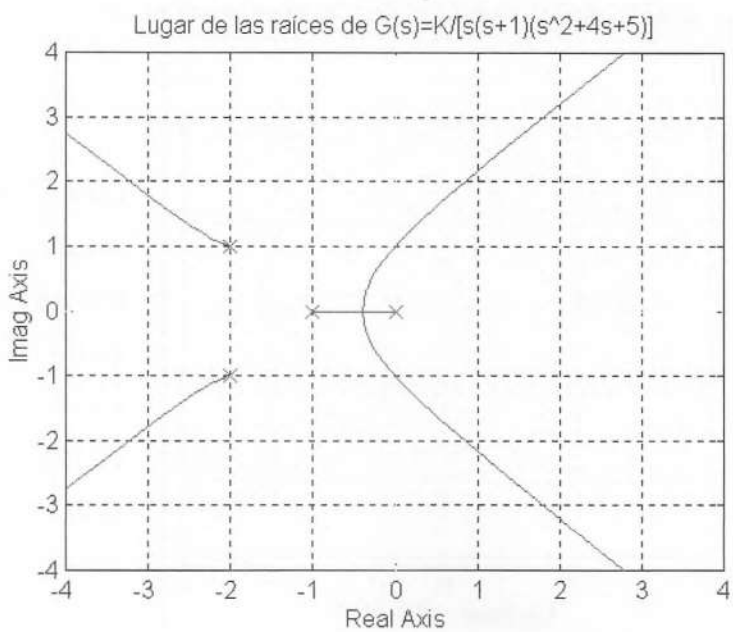


$$G(s) = \frac{K}{s(s+1)(s^2+4s+5)}$$

Programa en MATLAB

```
num=[0 0 0 0 1];
den=[1 5 9 5 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K/[s(s+1)(s^2+4s+5)]'
```

Lugar de las raíces



Ceros en lazo abierto

ninguno

Polos en lazo abierto

$s = -2 + j$

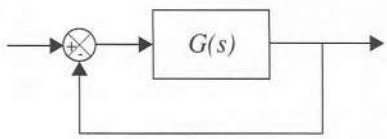
$s = -2 - j$

$s = -1$

$s = 0$

EJEMPLO 5-16

Sistema

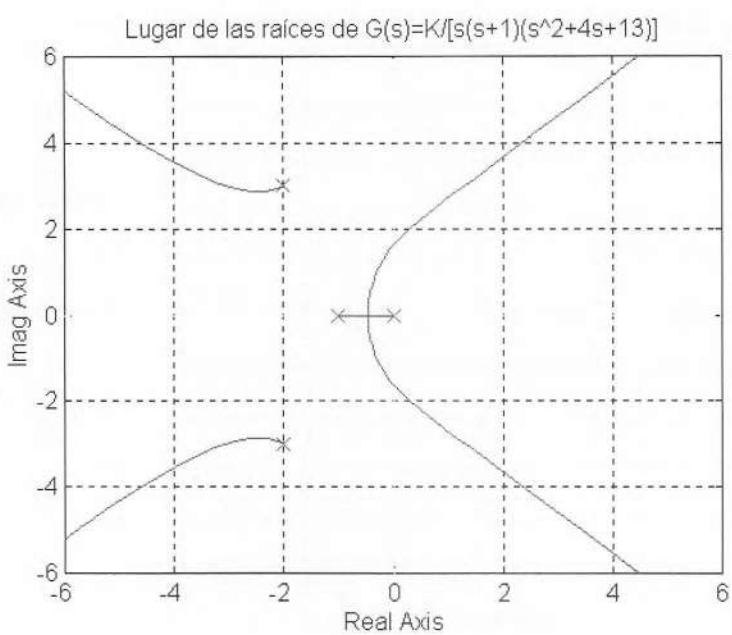


$$G(s) = \frac{K}{s(s+1)(s^2+4s+13)}$$

Programa en MATLAB

```
num=[0 0 0 0 1];
den=[1 5 17 13 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K/[s(s+1)(s^2+4s+13)]')
```

Lugar de las raíces



Ceros en lazo abierto

ninguno

Polos en lazo abierto

s = -2 + 3j

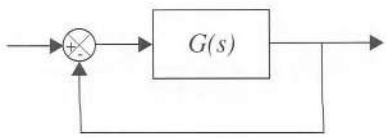
s = -2 - 3j

s = -1

s= 0

EJEMPLO 5-17

Sistema

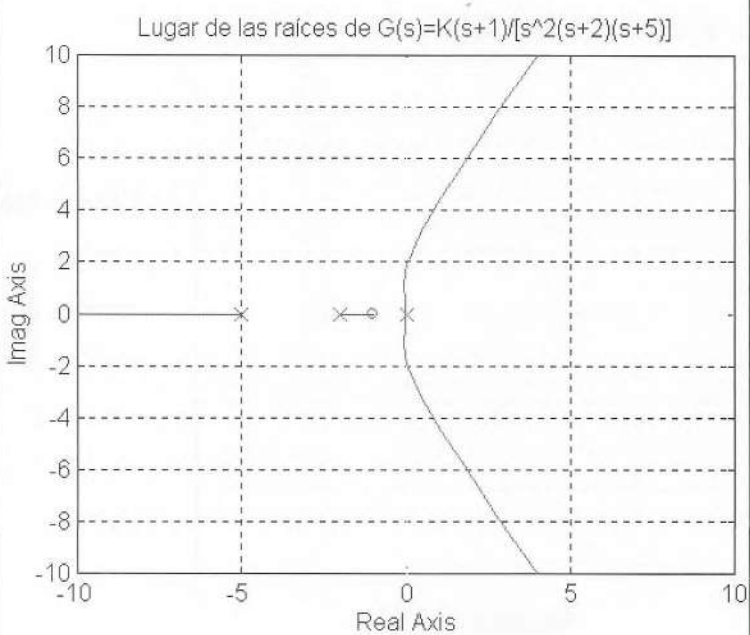


$$G(s) = \frac{K(s+1)}{s^2(s+2)(s+5)}$$

Programa en MATLAB

```
num=[0 0 0 1 1];
den=[1 7 10 0 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K(s+1)/[s^2(s+2)(s+5)]')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -1$

Polos en lazo abierto

$s = 0$

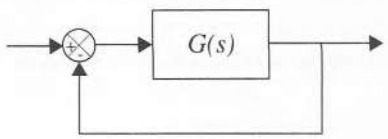
$s = 0$

$s = -2$

$s = -5$

EJEMPLO 5-18

Sistema

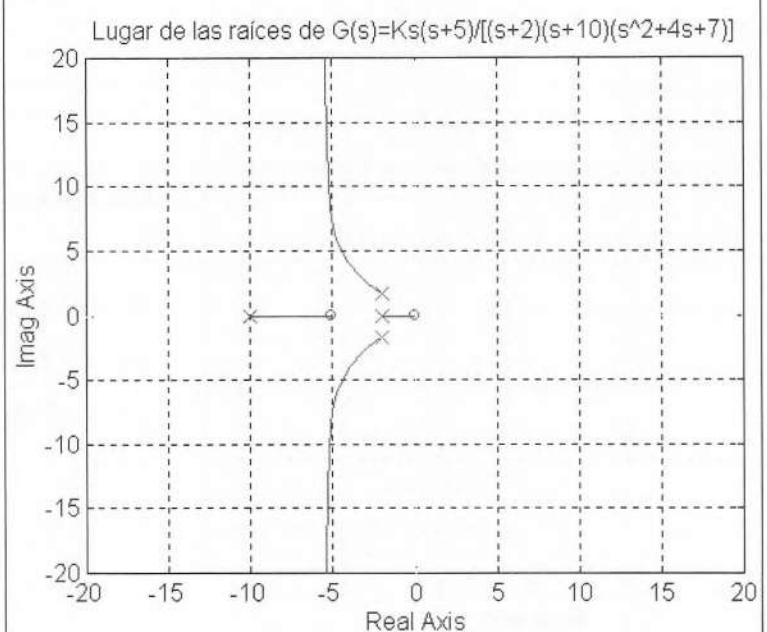


$$G(s) = \frac{Ks(s+5)}{(s+2)(s+5)(s^2+4s+7)}$$

Programa en MATLAB

```
num=[0 0 1 5 0];
den=[1 16 75 164 140];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=Ks(s+5)/[(s+2)(s+10)(s^2+4s+7)]')
```

Lugar de las raíces



Ceros en lazo abierto

$s = 0$

$s = -5$

Polos en lazo abierto

$s = -2 + 1.7321j$

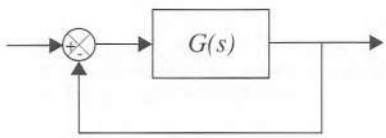
$s = -2 - 1.7321j$

$s = -2$

$s = -10$

EJEMPLO 5-19

Sistema

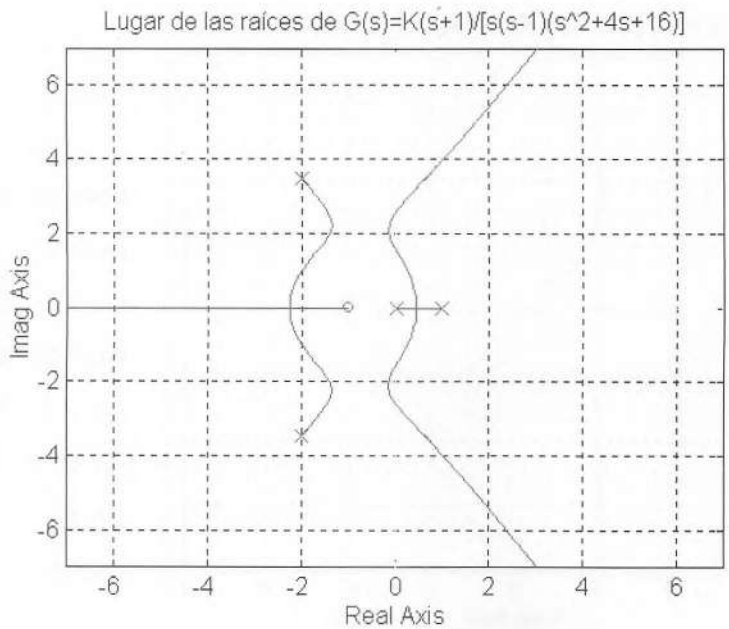


$$G(s) = \frac{K(s+1)}{s(s-1)(s^2+4s+16)}$$

Programa en MATLAB

```
den=[1 3 12 -16 0];

%*****Para conseguir que la región de representación gráfica sea cuadrada
%escriba la orden axis('square'). Con esta orden una línea con pendiente 1
%está a 45 grados y no girada debida a la forma irregular de la pantalla. Es
%importante saber que una gráfica que se imprima en papel puede ser o no
%cuadrada dependiendo de la impresora*****

axis('square')
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K(s+1)/[s(s-1)(s^2+4s+16)]')

%*****Para restaurar la relación entre los ejes a su valor normal
%introduzca la orden axis('normal')*****

axis('normal')
```

Lugar de las raíces



Ceros en lazo abierto

$s = -1$

Polos en lazo abierto

$s = -2 + 3.4641j$

$s = -2 - 3.4641j$

$s = 0$

$s = 1$

5.3 CASOS ESPECIALES

Para dibujar un lugar de las raíces, MATLAB calcula los polos en lazo cerrado utilizando únicamente un número razonablemente pequeño de valores de ganancia. Luego conecta los polos con una línea continua.

Considere el sistema en lazo cerrado mostrado en la Figura 5-14. Suponga que las funciones de transferencia $G(s)$ y $H(s)$ vienen dadas respectivamente como

$$G(s) = \frac{K(s+9)}{s(s^2+4s+11)}, \qquad H(s) = 1$$

El programa de MATLAB 5-7 generará un lugar de las raíces como el mostrado en la Figura 5-15. Obsérvese que los algoritmos automáticos de paso fijo del lugar de las raíces tienen pasos mayores donde la solución es más sensible al cambio.

En la Figura 5-15 las marcas 'o' dibujadas muestran los puntos donde el computador tiene solución. Si estos puntos se conectan con líneas continuas, se obtiene el lugar de las raíces tradicional que consiste en líneas o curvas continuas como muestra la Figura 5-16.

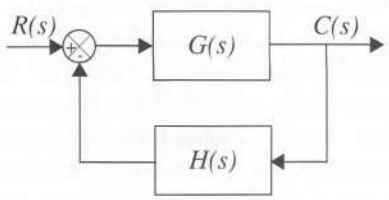


Figura 5-14

Programa en MATLAB 5-7

```
num=[0 0 1 9];
den=[1 4 11 0];
v=[-15 10 -15 15];
axis(v);
r=rlocus(num,den);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K(s+9)/[s(s^2+4s+11)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

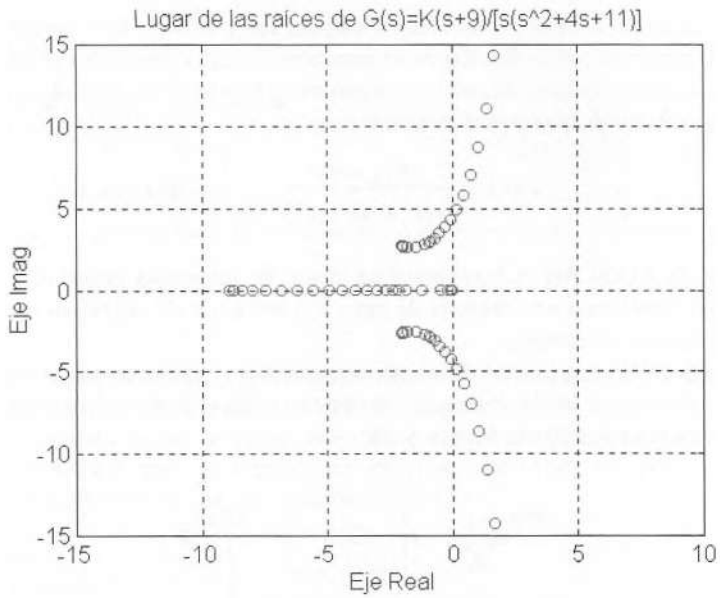


Figura 5-15

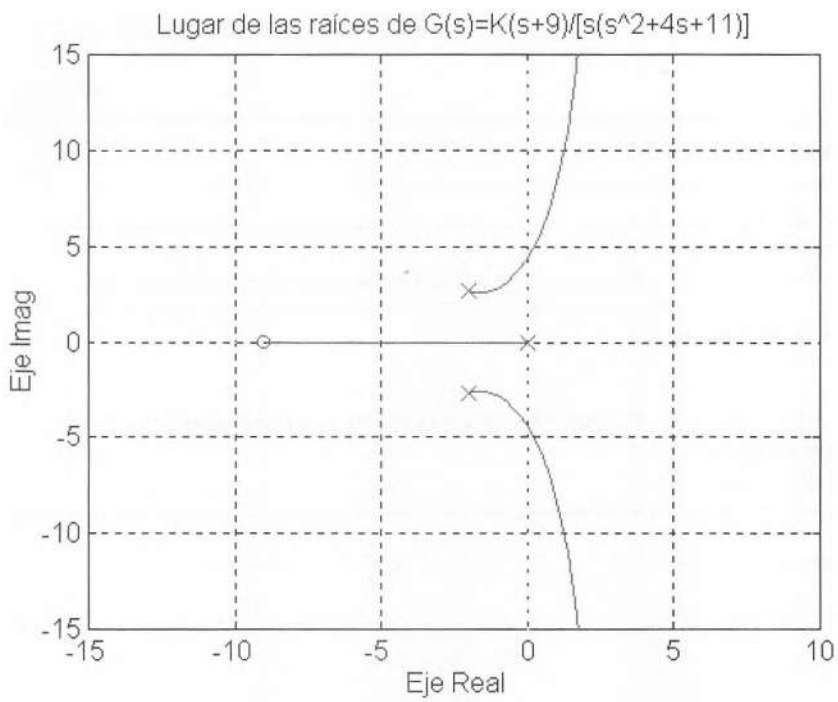


Figura 5-16

EJEMPLO 5-20

Considere el sistema mostrado en la Figura 5-17. La función de transferencia en lazo abierto es

$$G(s) = \frac{K(s+0.4)}{s^2(s+3.6)}$$

Vamos a dibujar el lugar de las raíces utilizando el programa de MATLAB 5-8, el cual genera un lugar de las raíces como el mostrado en la Figura 5-18. El lugar de las raíces debe ser simétrico respecto al eje real. Sin embargo, la Figura 5-18 muestra lo contrario. Si hacemos un zoom en la región *x-y* a

```
[-2  1  -1.5  1.5]
```

y utilizamos el programa de MATLAB 5-9, obtenemos el gráfico mostrado en la Figura 5-19.

MATLAB sustituye su propio conjunto de valores de ganancia que son utilizados para calcular un lugar de las raíces. Lo hace mediante una rutina interna adaptativa de paso fijo. Sin embargo, en algunos sistemas pequeños cambios en la ganancia causan cambios drásticos en la localización de las raíces dentro de algunos rangos de ganancias. Por tanto, MATLAB toma un salto grande en sus valores de ganancia cuando calcula las raíces, y la localización de las raíces cambia en una cantidad relativamente grande. Al dibujar, MATLAB conecta estos puntos y se obtiene un gráfico extraño donde se localizan las ganancias sensibles. Estos gráficos de lugares de las raíces erróneos ocurren normalmente cuando el lugar se aproxima a un polo doble (o triple o un polo de orden superior) ya que el lugar de las raíces es muy sensible a pequeños cambios en la ganancia.

Para arreglar estas figuras erróneas, es necesario especificar los valores de la ganancia K con un espaciado muy pequeño en el área donde las raíces cambian de forma rápida. Por ejemplo, si el rango está mal condicionado alrededor de $K = 8$, utilice un paso fijo pequeño alrededor de $K = 8$, utilice un paso fijo muy pequeño alrededor de $K = 8$. Defina K como

```
K1 = [0:0.1:7.5];
K2 = [7.5:0.005:8.5];
K3 = [8.5:1:100];
K = [K1  K2  K3];
```

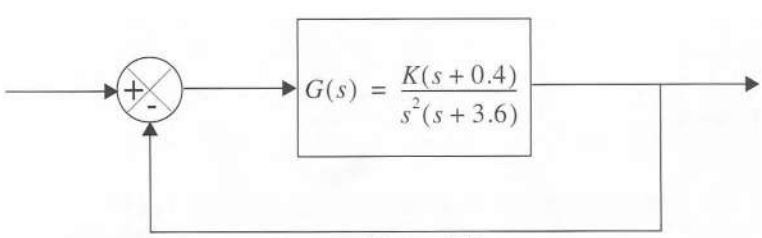


Figura 5-17

Programa en MATLAB 5-8

```
num=[0 0 1 0.4]; den=[1 3.6 0 0];
v=[-5 5 -5 5]; axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
```

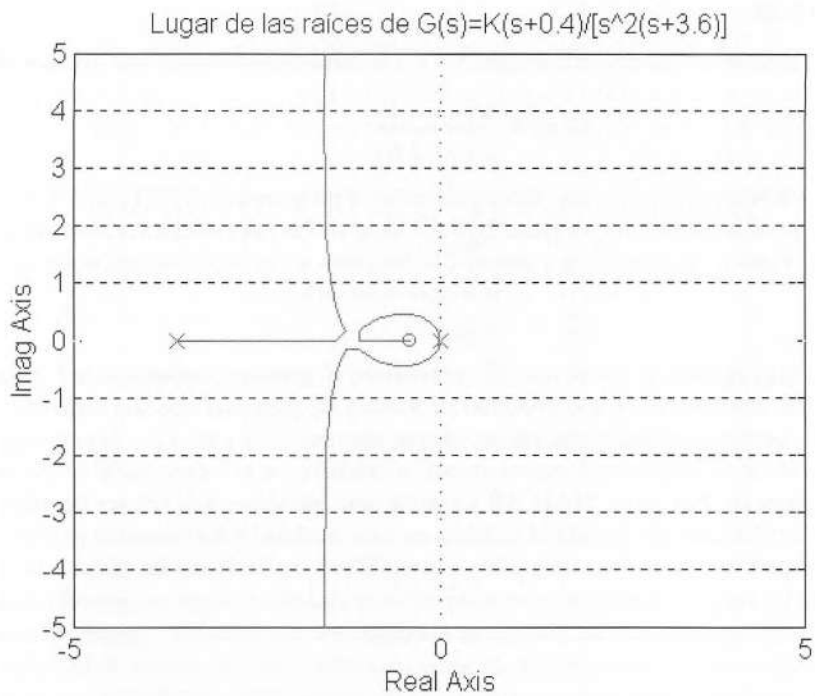


Figura 5-18

Programa en MATLAB 5-9

```
num=[0 0 1 0.4];
den=[1 3.6 0 0];
v=[-2 1 -1.5 1.5];
axis(v);
r=rlocus(num,den);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

y utilice la siguiente orden:

$$r = rlocus(num,den,K)$$

La Figura 5-20 tiene el mismo lugar de las raíces que el de la Figura 5-19, pero el primero está dibujado utilizando las marcas 'o'. Obsérvese que no se han calculados puntos próximos al punto (-1.25,0). Para darse cuenta de casos como éste, es aconsejable dibujar el lugar de las raíces utilizando las marcas 'o' o 'x'.

En general, para encontrar la región de ganancia K donde ocurre el problema, utilice una aproximación por tanteo. Por ejemplo, puede fijar la ganancia K como

$$K = 0:0.5:2$$

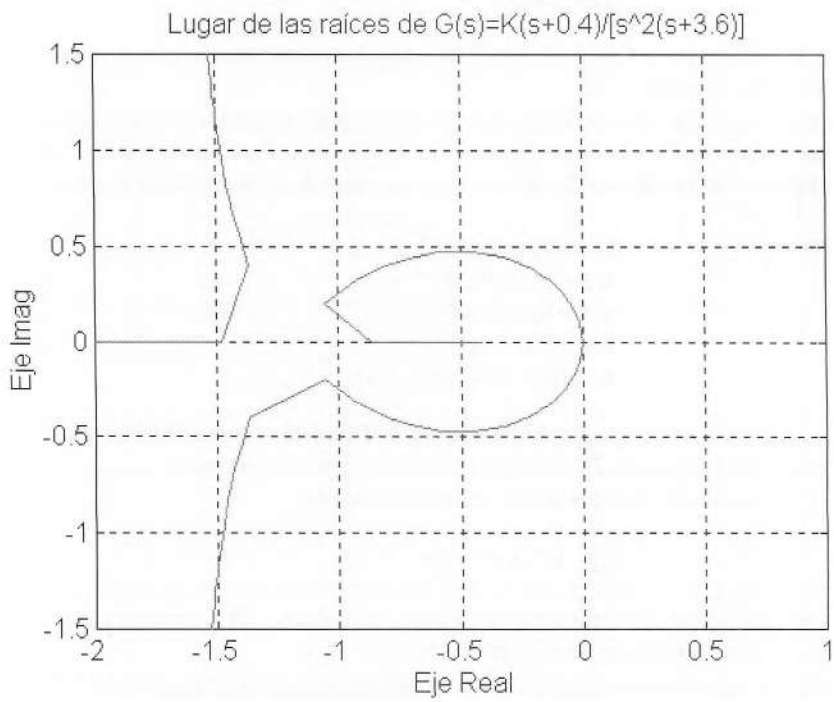


Figura 5-19

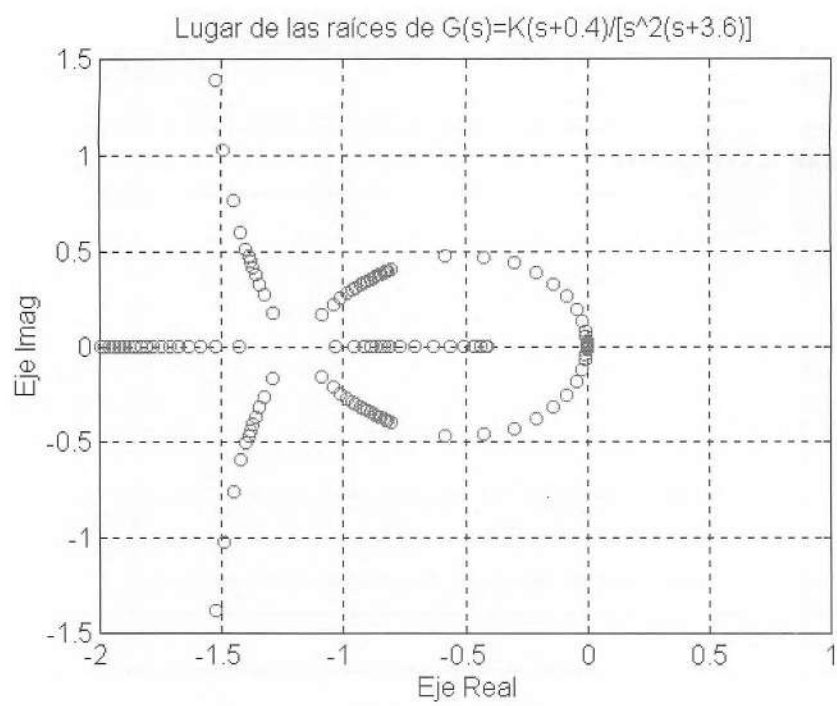


Figura 5-20

y observe dónde están localizadas las raíces correspondientes a $K = 2$. Después cambie la ganancia K a

```
K = 2:0.5:4
```

Localice las raíces para $K = 2$, $K = 4$ y así sucesivamente, encuentre el rango de ganancia K donde se necesita un paso fijo pequeño.

En el ejemplo que aquí se considera, la región crítica de la ganancia K se encuentra entre 3 y 5. Por tanto se necesita que el paso fijo sea bastante pequeño en esta región. Para las regiones $0 \leq K \leq 3$ y $5 \leq K \leq \infty$, el paso fijo debe ser mayor. Así, podemos dividir las regiones para K de la siguiente manera:

```
K1 = [0:0.2:3];
K2 = [3:0.01:5];
K3 = [5:0.2:10];
K4 = [10:5:200];
K = [K1  K2  K3  K4];
```

Vamos a introducir el programa de MATLAB 5-10 en el computador. La gráfica obtenida de este programa se muestra en la Figura 5-21. El lugar de las raíces parece correcto.

La Figura 5-22 muestra el lugar de las raíces en la región

```
[-2  1  -1.5  1.5]
```

Los círculos pequeños ('o') muestran los puntos calculados. El programa utilizado para obtener esta gráfica se muestra en el programa de MATLAB 5-11.

Al cambiar los pasos de la ganancia, como se muestra en el programa de MATLAB 5-12, se obtiene una ligera diferencia en el dibujo del lugar de las raíces, como se muestra en la Figura 5-23. Los dibujos de las Figuras 5-22 y 5-23 parecen correctos.

Un cambio en la región x-y, como el del programa de MATLAB 5-13, produce el dibujo mostrado en la Figura 5-24. Utilizando las marcas '-' obtenemos un lugar de las raíces más suave. (Véase el programa de MATLAB 5-14 y la Figura 5-25). La figura 5-25 es prácticamente igual a la Figura 5-21 que se había obtenido con anterioridad.

Programa en MATLAB 5-10

```
num=[0 0 1 0.4];
den=[1 3.6 0 0];
v=[-5 5 -5 5];
axis(v);
K1=[0:0.2:3];
K2=[3:0.01:5];
K3=[5:0.2:10];
K4=[10:5:200];
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r)
grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

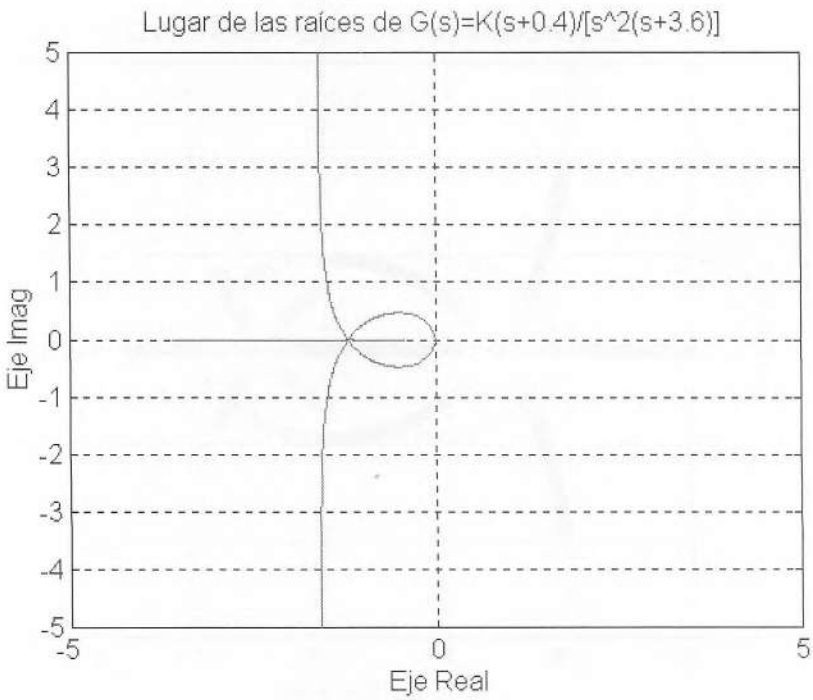


Figura 5-21

Programa en MATLAB 5-11

```
num=[0 0 1 0.4];
den=[1 3.6 0 0];
v=[-2 1 -1.5 1.5];
axis(v);
K1=[0:0.2:3];
K2=[3:0.01:5];
K3=[5:0.2:10];
K4=[10:5:200];
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

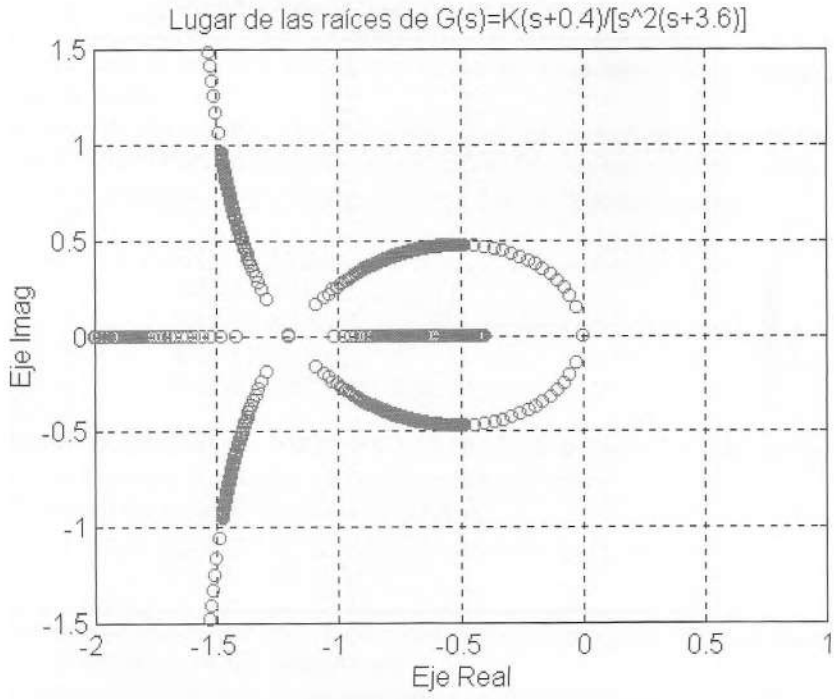


Figura 5-22

Programa en MATLAB 5-12

```
num=[0 0 1 0.4];
den=[1 3.6 0 0];
v=[-2 1 -1.5 1.5];
axis(v);
K1=[0:0.2:4.2];
K2=[4.2:0.002:4.4];
K3=[4.4:0.2:10];
K4=[10:5:200];
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

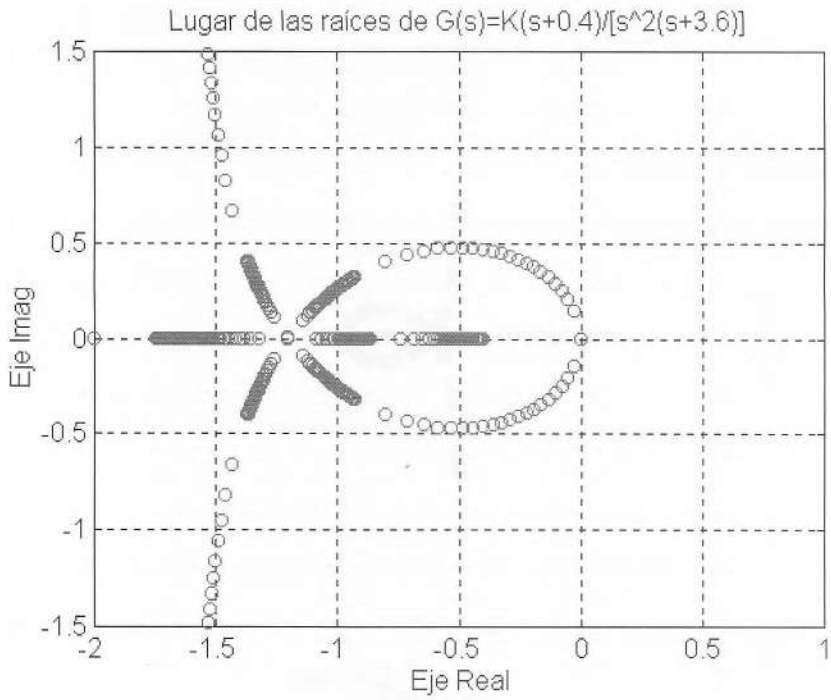


Figura 5-23

Programa en MATLAB 5-13

```
num=[0 0 1 0.4];
den=[1 3.6 0 0];
v=[-5 5 -5 5];
axis(v);
K1=[0:0.2:4.2];
K2=[4.2:0.002:4.4];
K3=[4.4:0.2:10];
K4=[10:5:200];
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

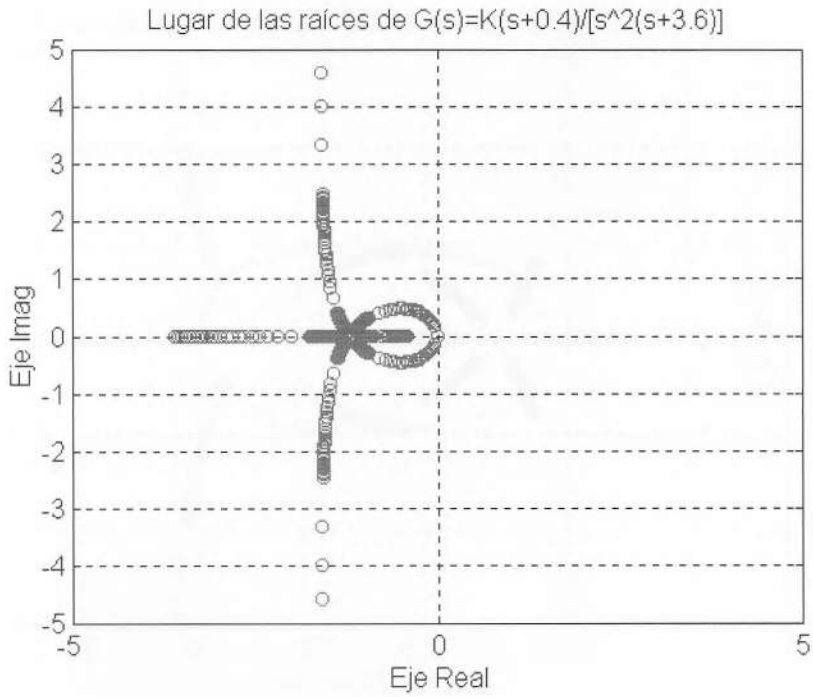


Figura 5-24

Programa en MATLAB 5-14

```
num=[0 0 1 0.4];
den=[1 3.6 0 0];
v=[-5 5 -5 5];
axis(v);
K1=[0:0.2:4.2];
K2=[4.2:0.002:4.4];
K3=[4.4:0.2:10];
K4=[10:5:200];
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'-')
grid
title('Lugar de las raíces de G(s)=K(s+0.4)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

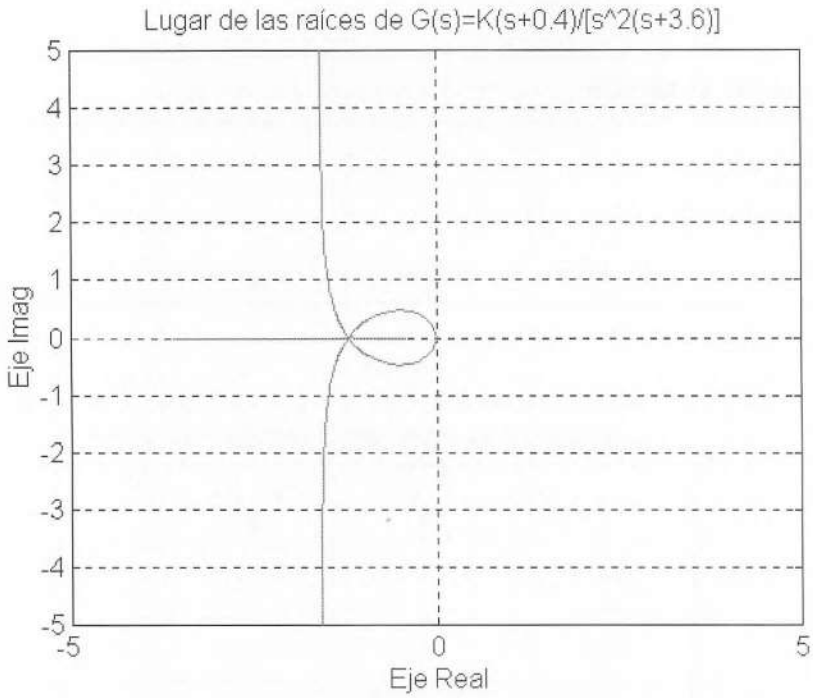


Figura 5-25

EJEMPLO 5-21

Considere el sistema mostrado en la Figura 5-26. La función de transferencia en lazo abierta es

$$G(s) = \frac{K}{(s-1)(s^2+4s+7)} = \frac{K}{s^3+3s^2+3s-7}$$

No hay ceros en lazo abierto. Los polos en lazo abierto están localizados en $s = -2 + j1.7321$, $s = -2 - j1.7321$ y $s = 1$. El programa de MATLAB 5-15 generará el lugar de las raíces mostrado en la Figura 5-27. [Obsérvese que dependiendo de la versión de MATLAB (como pueden ser la versión de estudiante o la versión profesional 4.0) se pueden obtener lugares de las raíces ligeramente diferentes].

En este gráfico, observe los cambios bruscos en la dirección de las ramas del lugar de las raíces. Tales cambios dan a entender que hay un problema en el dibujo del lugar de las raíces. Para investigar el problema, se puede dibujar el lugar de las raíces con las marcas 'o' y 'x'. Utilizando el programa de MATLAB 5-16 se obtiene el lugar de las raíces mostrado en la Figura 5-28. Se puede ver que no se han calculado puntos cerca del punto (-1,0), y ésta es la causa del problema.

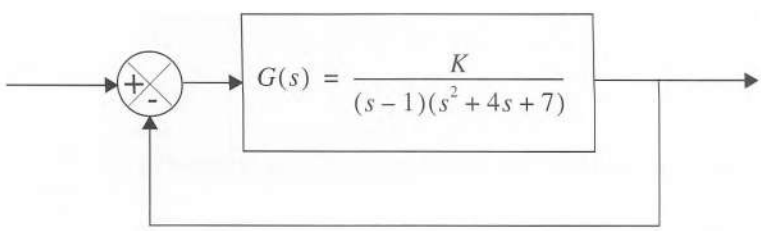


Figura 5-26

Programa en MATLAB 5-15

```
num=[0 0 0 1];
den=[1 3 3 -7];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K/[(s-1)(s^2+4s+7)]')
```

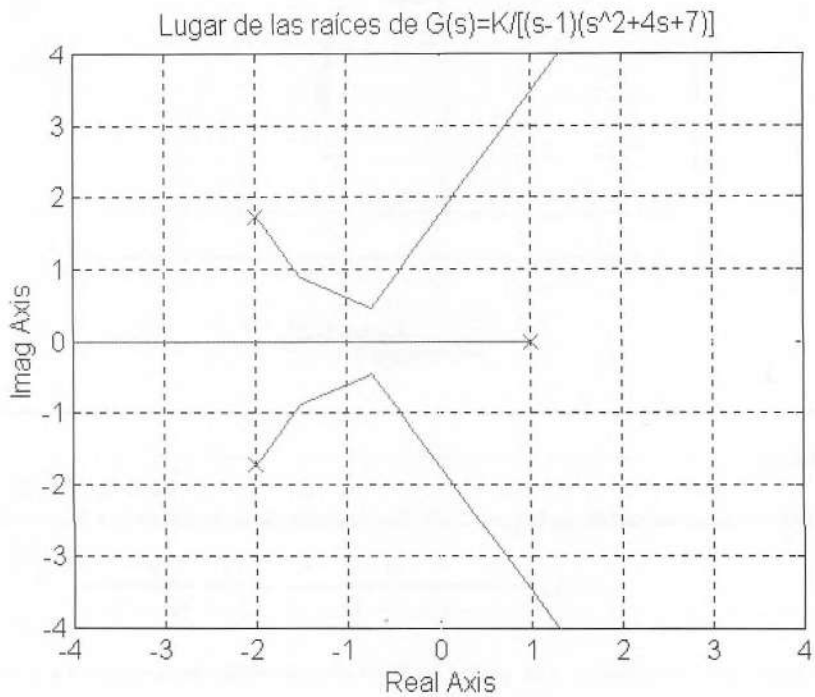


Figura 5-27

Programa en MATLAB 5-16

```
num=[0 0 0 1];
den=[1 3 3 -7];
v=[-4 4 -4 4];
axis(v);
r=rlocus(num,den);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K/[(s-1)(s^2+4s+7)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

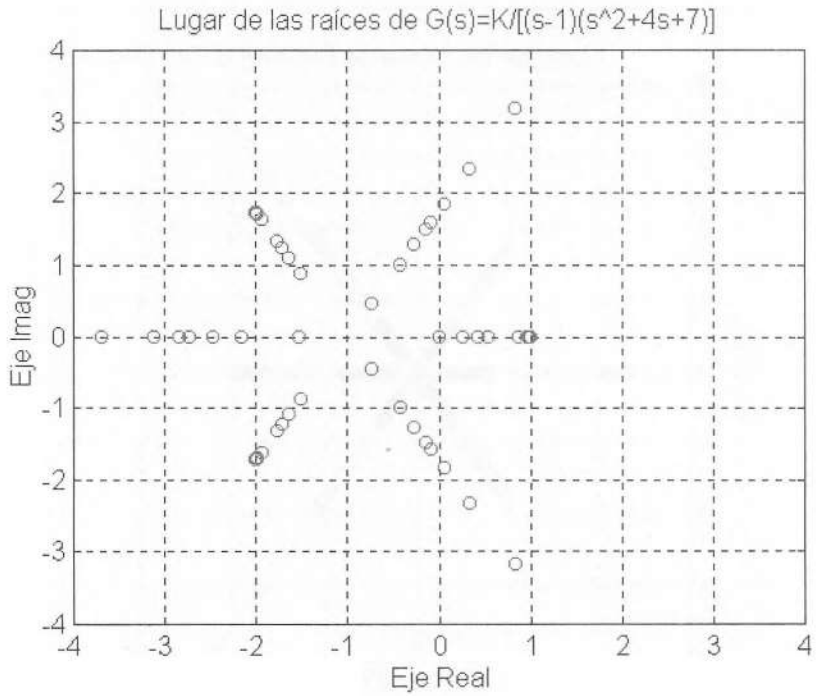


Figura 5-28

Utilizando una aproximación por tanteo cambiando la ganancia K entre $0 \leq K \leq 2$, $2 \leq K \leq 4$, y así sucesivamente se puede encontrar la región crítica [incluido el punto (-1,0)] entre $7{,}5 \leq K \leq 8{,}5$. Por tanto, haciendo el paso fijo menor en esta región crítica como muestra el programa de MATLAB 5-17 se obtiene el dibujo del lugar de las raíces mostrado en la Figura 5-29.

Si se cambia la orden plot(y,'o') por plot(y,'-'), como muestra el programa de MATLAB 5-18, se obtiene el lugar de las raíces normal, como se observa en la Figura 5-30, el cual es correcto. (El lugar de las raíces para este sistema consiste de tres líneas rectas).

Programa en MATLAB 5-17

```
num=[0 0 0 1];
den=[1 3 3 -7];
v=[-4 4 -4 4];
axis(v);
K1=0:0.5:7.5;
K2=7.5:0.01:8.5;
K3=8.5:1:20;
K4=20:10:100;
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'o')
grid
title('Lugar de las raíces de G(s)=K/[(s-1)(s^2+4s+7)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

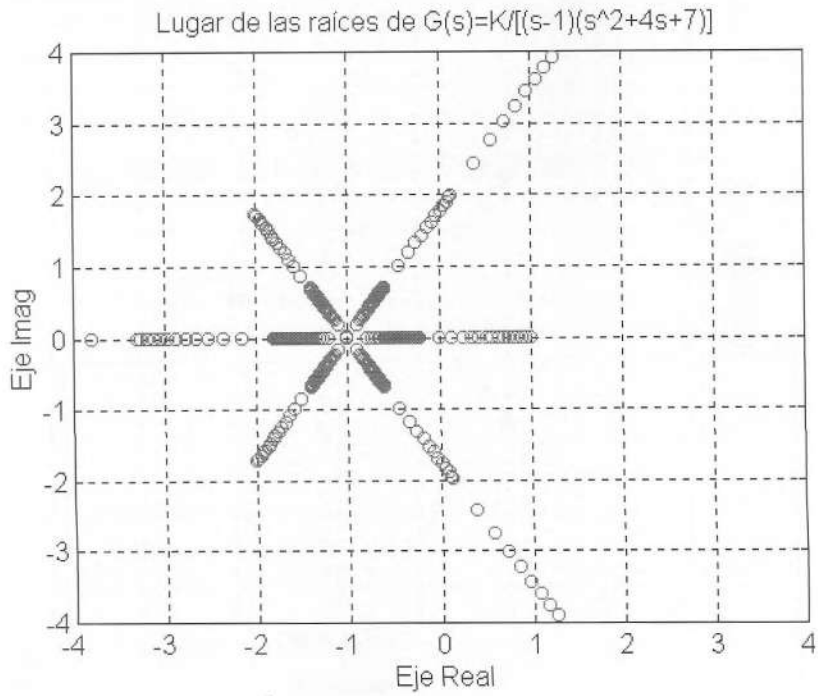


Figura 5-29

Programa en MATLAB 5-18

```
num=[0 0 0 1];
den=[1 3 3 -7];
v=[-4 4 -4 4];
axis(v);
K1=0:0.5:7.5;
K2=7.5:0.01:8.5;
K3=8.5:1:20;
K4=20:10:100;
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'-')
grid
title('Lugar de las raíces de G(s)=K/[(s-1)(s^2+4s+7)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

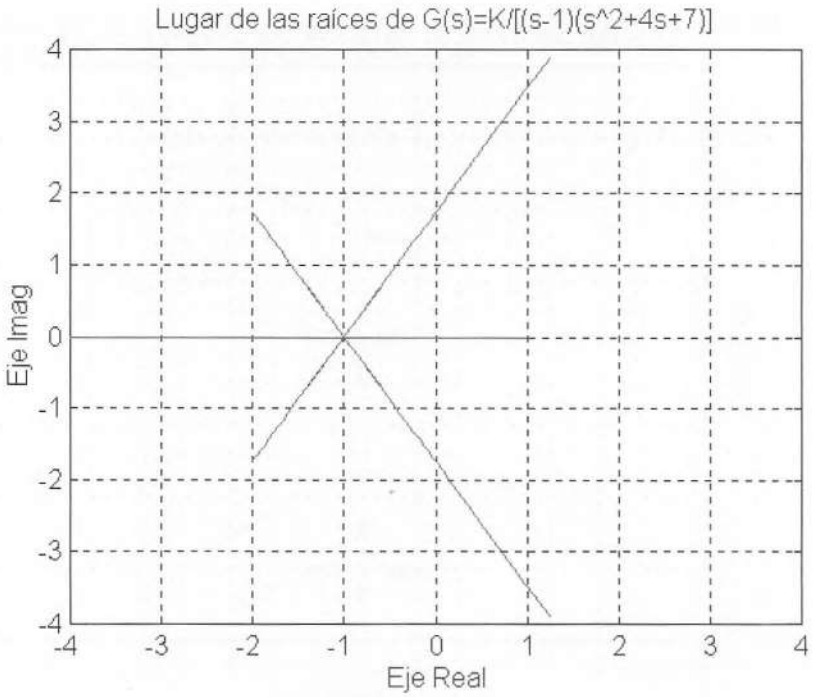


Figura 5-30

EJEMPLO 5-22

Considere un sistema cuya función de transferencia en lazo abierto $G(s)H(s)$ viene dada por

$$G(s)H(s) = \frac{K}{(s^2 + 2s + 2)(s^2 + 2s + 5)}$$

$$= \frac{K}{s^4 + 4s^3 + 11s^2 + 14s + 1(}$$

No existen ceros en lazo abierto. Los polos del lazo abierto están en $s = -1 + j$, $s = -1 - j$, $s = -1 + 2j$ y $s = -1 - 2j$.

El programa de MATLAB 5-19 generará el lugar de las raíces que se muestra en la Figura 5-31. En este lugar de las raíces, se observa cómo dos ramas en el semiplano superior (y también en el semiplano inferior) se aproximan una hacia la otra pero sin tocarse. Si se inspecciona cuidadosamente este diagrama se observa que existen dos segmentos rectos cortos en el semiplano superior (y en el inferior). Estos segmentos rectos cortos se pueden deber a un escaso número de puntos calculados. Para comprobarlo se puede introducir el programa de MATLAB 5-20 y dibujar el lugar de las raíces utilizando las marcas 'o'.

El gráfico se muestra en la Figura 5-32.

Programa en MATLAB 5-19

```
num=[0 0 0 0 1]; den=[1 4 11 14 10];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K/[(s^2+2s+2)(s^2+2s+5)]')
```

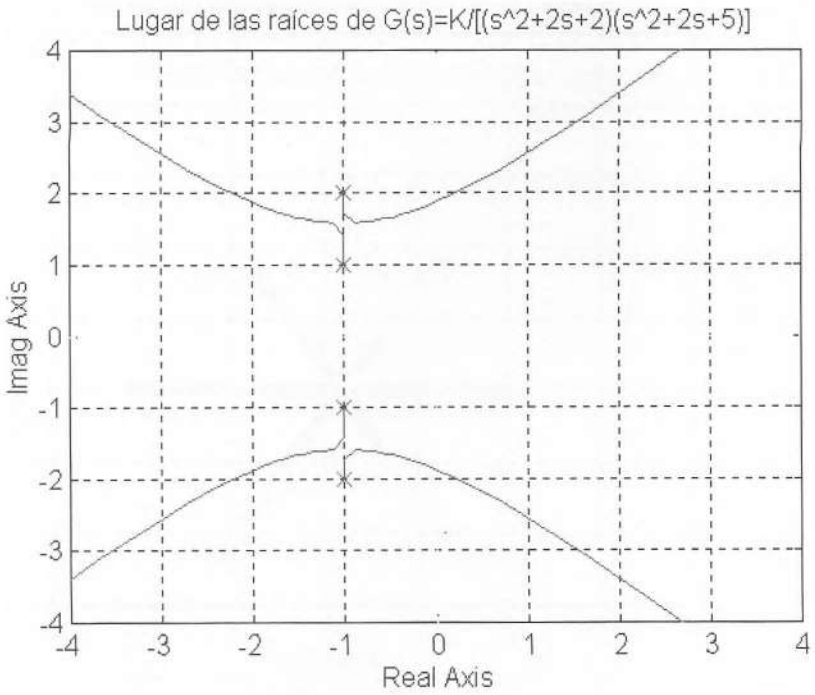


Figura 5-31

Para ampliar la región donde las ramas del lugar de las raíces deberían cruzarse entre sí, se van a elegir los ejes x e y como sigue:

$$2 \leq x \leq 0, \qquad -3 \leq y \leq 3$$

Con un tamaño de ganancia pequeño en la región $2{,}0 \leq K \leq 2{,}5$ se puede obtener un mejor dibujo del lugar de las raíces. Utilizando el programa de MATLAB 5-21 se obtiene el lugar de las raíces que se muestra en la Figura 5-33. Observe que no existen puntos calculados cerca del (-1, 1,6) y del (-1, -1,6). Las regiones cercanas a estos puntos necesitan ser estudiadas cuidadosamente.

Programa en MATLAB 5-20

```
num=[0 0 0 0 1];
den=[1 4 11 14 10];
v=[-4 4 -4 4];
axis(v);
r=rlocus(num,den);
plot(r,'og'), grid
title('Lugar de las raíces de G(s)=K/[(s^2+2s+2)(s^2+2s+5)]')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(r,'og')' visualiza en la pantalla pequeños
%círculos de color verde*****
```

Mediante una aproximación por tanteo, los valores de ganancia cercanos a estos puntos se encuentran en $2{,}0 \leq K \leq 2{,}5$. Con el programa de MATLAB 5-22 se obtiene el lugar de las raíces mostrado en la Figura 5-34.

Claramente las dos ramas del semiplano superior (y del inferior) se cruzan entre sí. Al ampliar la región de K a 800 y utilizando líneas y curvas continuas, como se muestra en el programa de MATLAB 5-23, se obtiene el dibujo que muestra la Figura 5-35. Este dibujo es correcto.

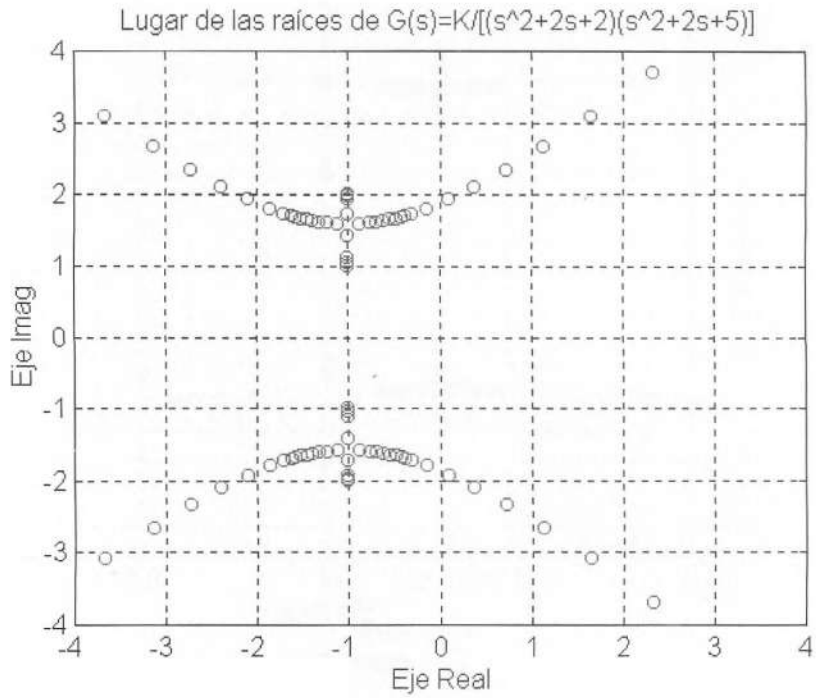


Figura 5-32

Programa en MATLAB 5-21

```
num=[0 0 0 0 1];
den=[1 4 11 14 10];
v=[-2 0 -3 3];
axis(v);
r=rlocus(num,den);
plot(r,'or')
grid
title('Lugar de las raíces de G(s)=K/[(s^2+2s+2)(s^2+2s+5)]')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(r,'or')' visualiza en la pantalla pequeños
%círculos de color rojo*****
```

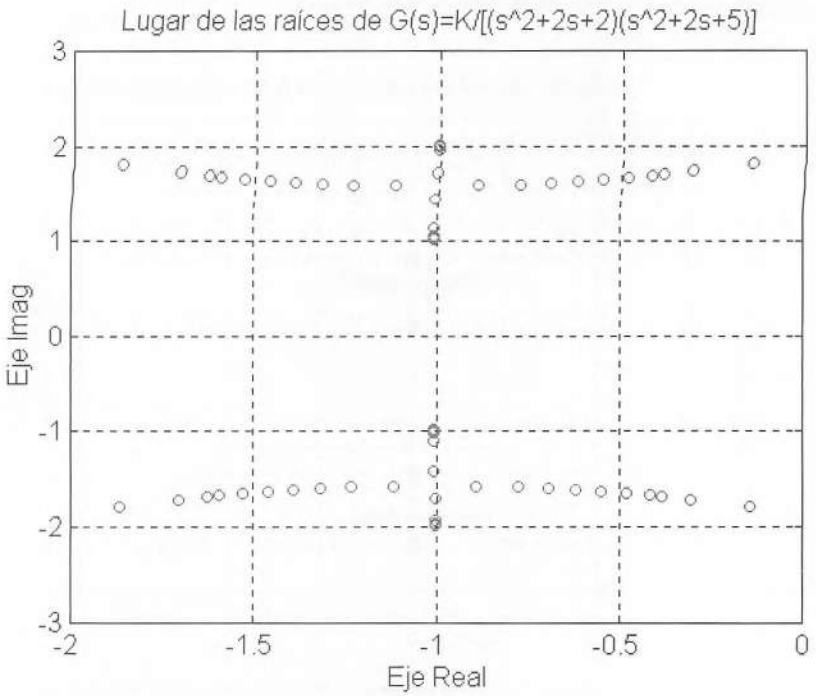


Figura 5-33

Programa en MATLAB 5-22

```
num=[0 0 0 0 1];
den=[1 4 11 14 10];
v=[-2 0 -3 3];
axis(v);
K1=0:0.5:2;
K2=2:0.01:2.5;
K3=2.5:0.5:10;
K4=10:10:200;
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
plot(r,'og')
grid
title('Lugar de las raíces de G(s)=K/[(s^2+2s+2)(s^2+2s+5)]')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(r,'og')' visualiza en la pantalla pequeños
%círculos de color verde*****
```

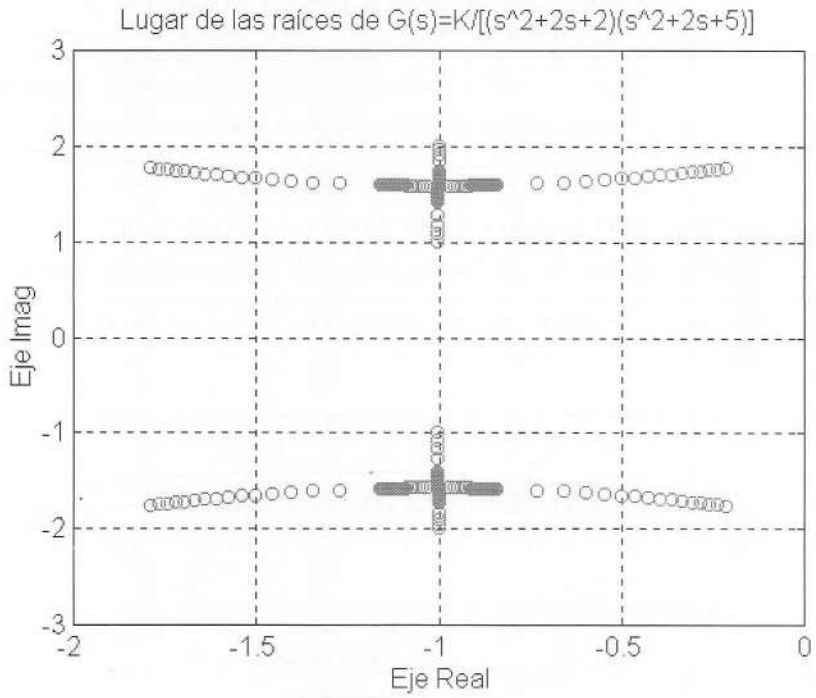


Figura 5-34

Programa en MATLAB 5-23

```
num=[0 0 0 0 1];
den=[1 4 11 14 10];
v=[-2 0 -3 3];
axis(v);
K1=0:0.5:2;
K2=2:0.01:2.5;
K3=2.5:0.5:10;
K4=10:10:50;
K5=50:50:800;
K=[K1 K2 K3 K4 K5];
r=rlocus(num,den,K);
plot(r,'-g')
grid
title('Lugar de las raíces de G(s)=K/[(s^2+2s+2)(s^2+2s+5)]')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(r,'-g')' visualiza en la pantalla pequeños
%círculos de color verde*****
```

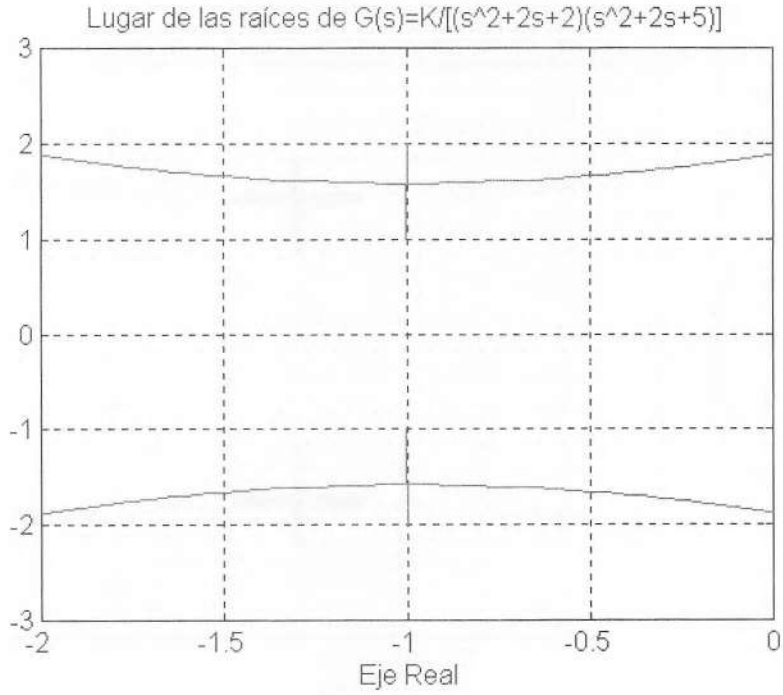


Figura 5-35

EJEMPLO 5-23

Considere el sistema cuya función de transferencia en lazo abierto $G(s)H(s)$ es

$$G(s)H(s) = \frac{K}{s(s+0{,}5)(s^2+0{,}6s+10)}$$

$$= \frac{K}{s^4+1{,}1s^3+10{,}3s^2+5s}$$

No existen ceros en lazo abierto. Los polos en lazo abierto están localizados en $s = -0{,}3 + j3{,}1480$, $s = -0{,}3 - j3{,}1480$, $s = -0{,}5$ y $s = 0$.

Al introducir el programa de MATLAB 5-24 en el computador, se obtiene el lugar de las raíces de la Figura 5-36.

Observe que en las regiones próximas a $x = -0{,}3$, $y = 2{,}3$ y $x = -0{,}3$, $y = -2{,}3$ dos lugares de las raíces se aproximan entre sí. Podemos preguntarnos si estas dos ramas deberían tocarse. Para examinar esta situación, podemos dibujar el lugar de las raíces con la orden

```
r = rlocus(num,den)
plot(r,'o')
```

como se muestra en el programa de MATLAB 5-25. La Figura 5-37 muestra la gráfica resultante.

Al no haber puntos calculados cerca de (-0,3, 2,3) y (-0,3, -2,3), es necesario ajustar los pasos en la ganancia K. Mediante una aproximación por tanteo, se encuentra una región de interés entre $20 \leq K \leq 30$. Al introducir el programa de MATLAB 5-26 se obtiene el lugar de las raíces de la Figura 5-38. De este dibujo se ve claro que las dos ramas que se aproximan en el semiplano superior (o en el semiplano inferior) no se tocan.

Programa en MATLAB 5-24

```
num=[0 0 0 0 1];
den=[1 1.1 10.3 5 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K/[s(s+0.5)(s^2+0.6s+10)]')
```

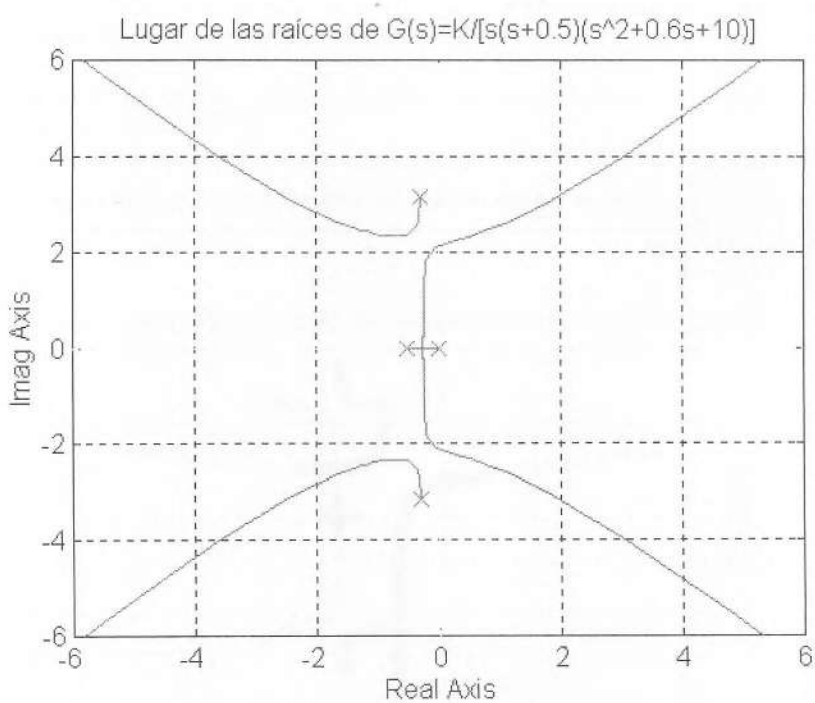


Figura 5-36

Programa en MATLAB 5-25

```
num=[0 0 0 0 1];
den=[1 1.1 10.3 5 0];
v=[-6 6 -6 6];
axis(v);
r=rlocus(num,den);
plot(r,'or')
grid
title('Lugar de las raíces de G(s)=K/[s(s+0.5)(s^2+0.6s+10)]')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(r,'or')' visualiza en la pantalla pequeños
%círculos de color rojo*****
```

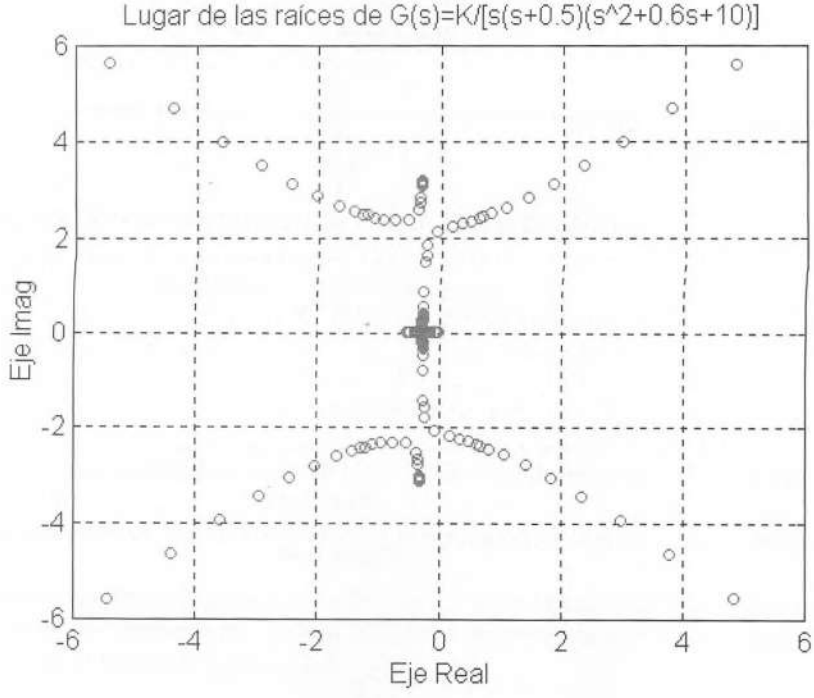


Figura 5-37

Programa en MATLAB 5-26

```
num=[0 0 0 0 1];
den=[1 1.1 10.3 5 0];
v=[-2 2 -4 4];
axis(v);
K1=0:0.2:20;
K2=20:0.1:30;
K3=30:5:100;
K=[K1 K2 K3];
r=rlocus(num,den,K);
plot(r,'ob')
grid
title('Lugar de las raíces de G(s)=K/[s(s+0.5)(s^2+0.6s+10)]')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(r,'ob')' visualiza en la pantalla pequeños
%círculos de color azul*****
```

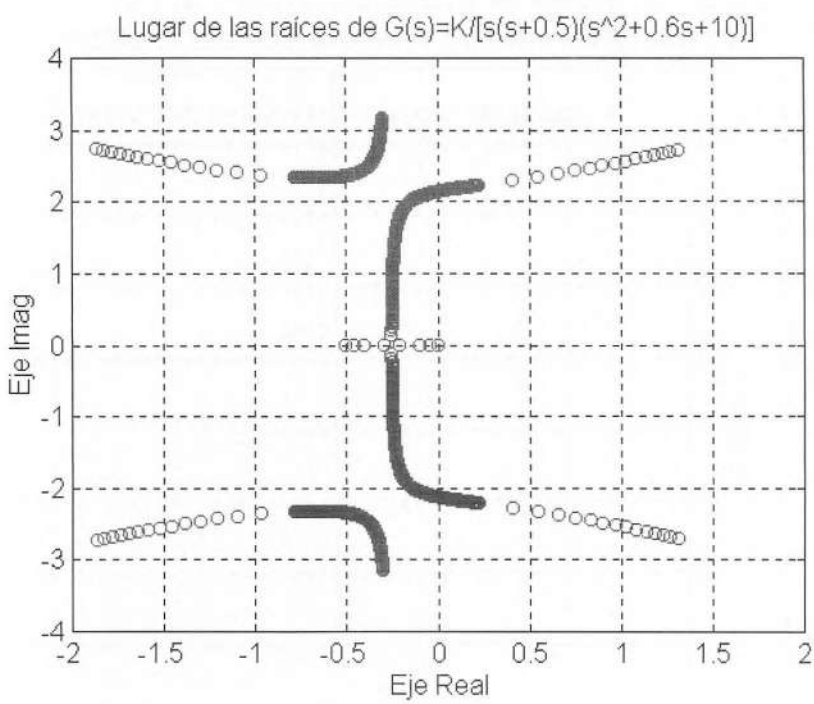


Figura 5-38

EJEMPLO 5-24

Considere el sistema cuya función de transferencia en lazo abierto $G(s)H(s)$ es

$$G(s)H(s) = \frac{K(s+0{,}2)}{s^2(s+3{,}6)}$$

El cero en lazo abierto está en $s = -0{,}2$, y los polos en lazo abierto están localizados en $s = 0$, $s = 0$ y $s = -3{,}6$. Al introducir el programa de MATLAB 5-27 en el computador obtendrá el lugar de las raíces de la Figura 5-39. El lugar de las raíces generado por el computador parece correcto.

Programa en MATLAB 5-27

```
num=[0 0 1 0.2];
den=[1 3.6 0 0];
v=[-4 2 -4 4];
axis(v);
rlocus(num,den);

Auto scales auto-ranged

Auto scales frozen

grid
title('Lugar de las raíces de G(s)=K(s+0.2)/[s^2(s+3.6)]')
```

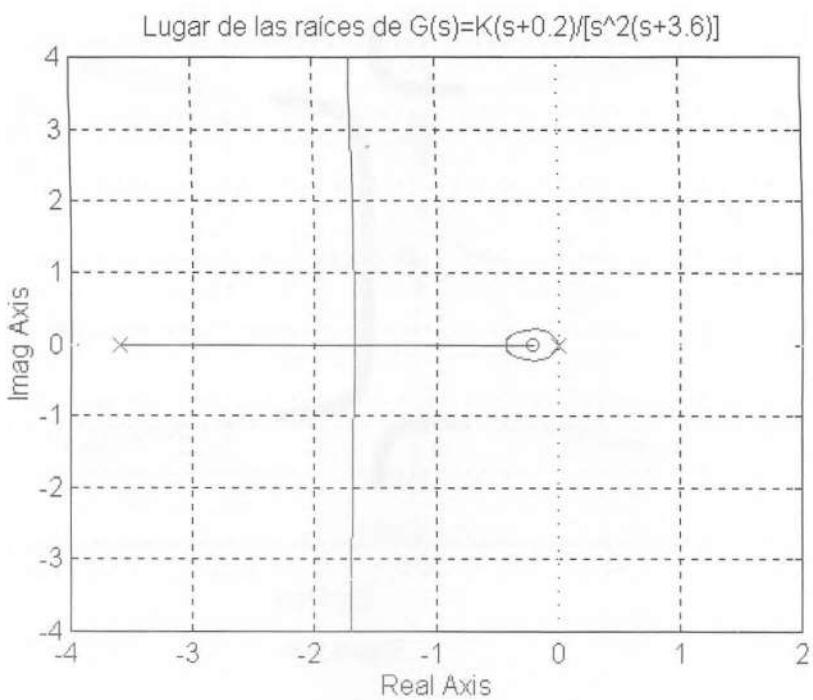


Figura 5-39

En los ejemplos del 5-20 al 5-23, se discute cómo solucionar los problemas del lugar de las raíces causados por un incremento inadecuado en el tamaño de la ganancia. Por otra parte, utilizando un incremento inadecuado de la ganancia K es posible "engañar" a MATLAB y generar un dibujo del lugar de las raíces erróneo y extraño.

A continuación, generaremos un curioso lugar de las raíces utilizando un incremento constante de la ganancia K. (Calculamos los polos en lazo cerrado de este sistema para $K = 0, 1, 2, ..., 200$ y unimos los puntos calculados). El programa de MATLAB 5-28 producirá el lugar de las raíces que se muestra en la Figura 5-40. Obviamente es un dibujo del lugar de las raíces erróneo.

Programa en MATLAB 5-28

```
num=[0 0 1 0.2];
den=[1 3.6 0 0];
v=[-4 2 -4 4];
axis(v);
K=0:1:200;
r=rlocus(num,den,K);
plot(r)
grid
title('Lugar de las raíces erróneo de G(s)=K(s+0.2)/[s^2(s+3.6)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

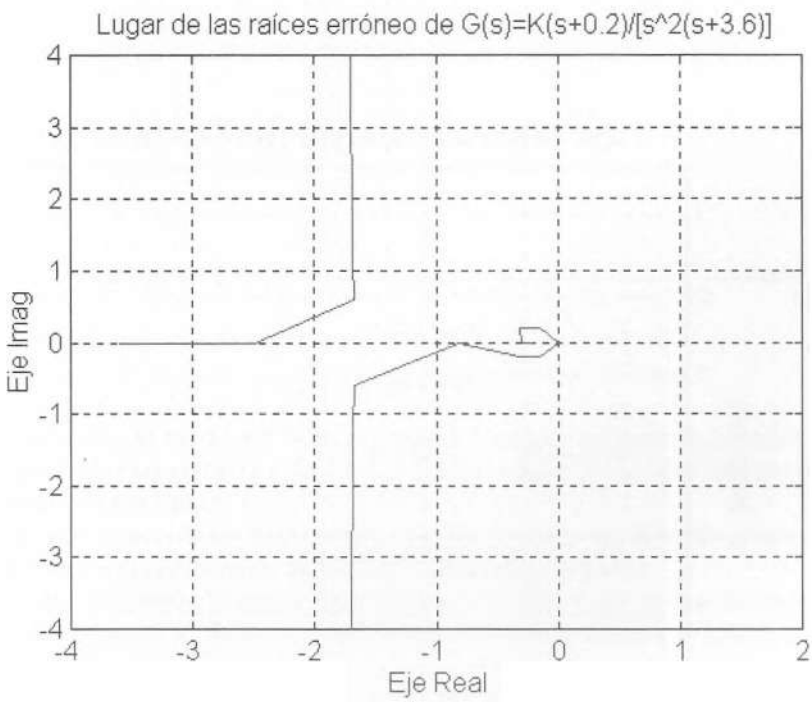


Figura 5-40

EJEMPLO 5-25

Considere un sistema cuya función de transferencia en lazo abierto es

$$G(s)H(s) = \frac{K(s+0{,}5)(s+3)}{(s+1)(s+2)}$$

Si se introduce el programa de MATLAB 5-29 en el computador, se obtiene el lugar de las raíces de la Figura 5-41. Este dibujo es un lugar de las raíces correcto.

Pero si utilizamos la orden

r = rlocus(num,den)

e introducimos el programa de MATLAB 5-30 en el computador, se generará un lugar de las raíces erróneo, como se observa en la Figura 5-42.

Si se especifica la región *x*-*y* como

[-5 15 -6 6]

Programa en MATLAB 5-29

```
num=[1 3.5 1.5];
den=[1 3 2];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K(s+0.5)(s+3)/[(s+1)(s+2)]')
```

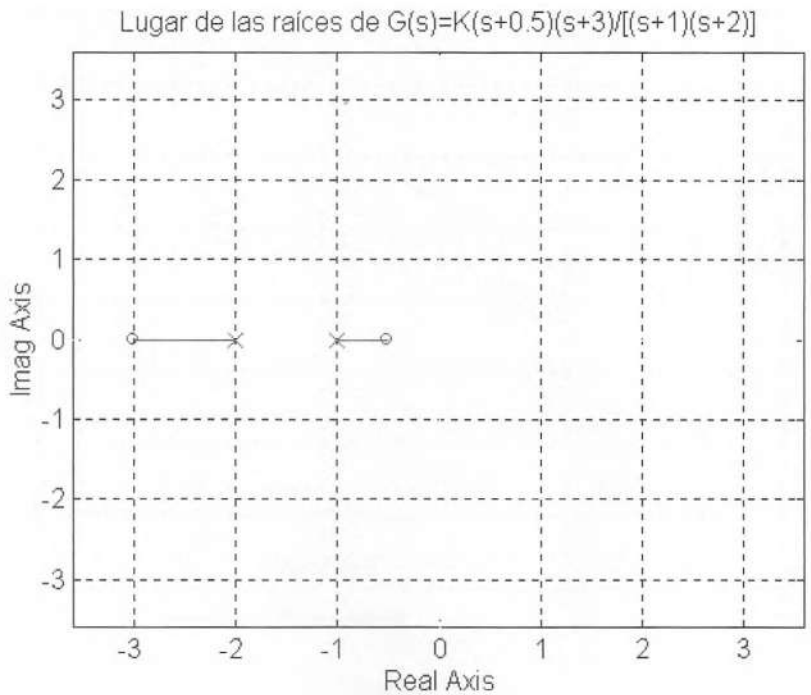


Figura 5-41

Programa en MATLAB 5-30

```
num=[1 3.5 1.5];
den=[1 3 2];
r=rlocus(num,den);
plot(r,'o')
grid
title('Lugar de las raíces erróneo')
xlabel('Eje Real')
ylabel('Eje Imag')
```

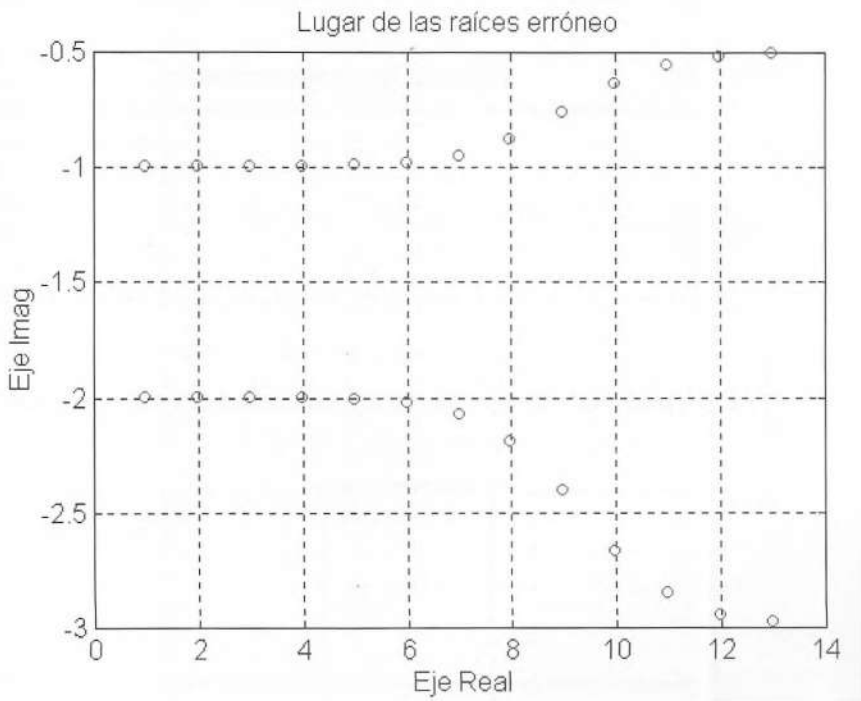


Figura 5-42

y se introduce el programa de MATLAB 5-31, se volverá a obtener un lugar de las raíces erróneo, como se muestra en la Figura 5-43. Este gráfico puede ocurrir cuando el lugar de las raíces no introduce la región compleja y únicamente está el Eje Real.

Para examinar estos lugares de las raíces erróneos, es necesario comprobar los valores de r. Para conseguir los valores de r introduzca el programa de MATLAB 5-32 en el computador.

De los valores de r obtenidos, se puede observar que son reales y que se encuentran entre -1 y -0,5070 o entre -2 y -2,9815. La Figura 5-43 muestra un gráfico de r frente a k, donde k = 1, 2, 3, ..., 13. (Es un gráfico incorrecto para el lugar de las raíces).

Programa en MATLAB 5-31

```
num=[1 3.5 1.5];
den=[1 3 2];
v=[-5 15 -6 6];
axis(v)
r=rlocus(num,den);
plot(r,'o')
grid
title('Lugar de las raíces erróneo')
xlabel('Eje Real')
ylabel('Eje Imag')
```

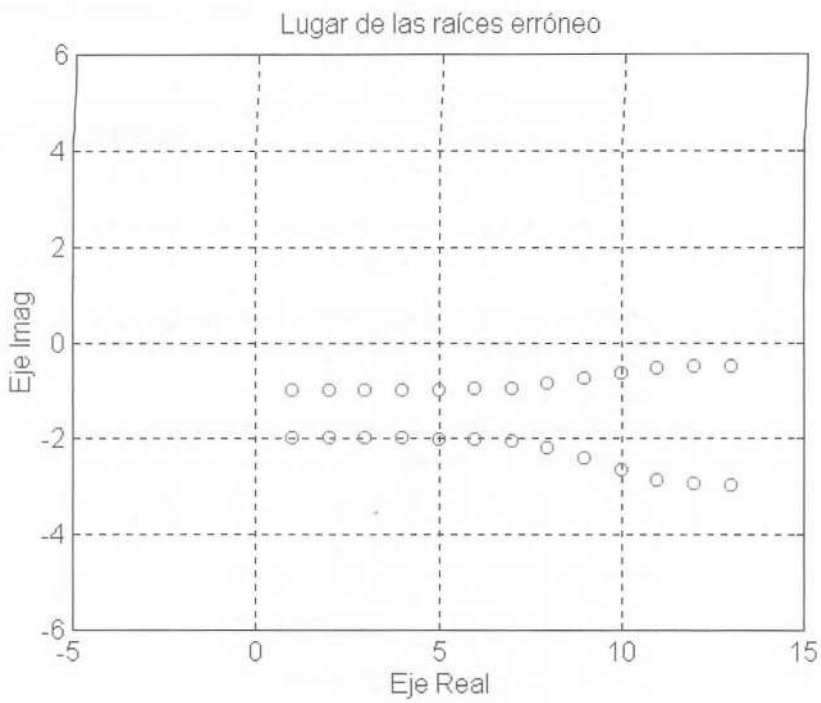


Figura 5-43

Para corregir esta situación, primero comprobamos el tamaño de la matriz r como sigue:

```
size(r)

ans =

    13     2
```

Después se introduce la siguiente orden plot:

plot(r,zeros(13,1))

Programa en MATLAB 5-32

```
num=[1 3.5 1.5];
den=[1 3 2];
v=[-5 15 -6 6];
axis(v)
r=rlocus(num,den)

r =

   -2.0000   -1.0000
   -2.0000   -1.0000
   -2.0007   -0.9995
   -2.0027   -0.9982
   -2.0088   -0.9942
   -2.0265   -0.9825
   -2.0755   -0.9509
   -2.1932   -0.8790
   -2.4086   -0.7603
   -2.6668   -0.6363
   -2.8551   -0.5564
   -2.9463   -0.5204
   -2.9815   -0.5070
```

donde zeros(13,1) es un vector columna que está formado por 13 ceros como sigue:

```
zeros(13,
1)

ans =

     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
     0
```

Esto significa que en lugar de dibujar los valores de r frente a k (k = 1, 2, 3, ..., 13), vamos a permitir a MATLAB que dibuje los ceros frente a r.

El programa de MATLAB 5-33 da un dibujo del lugar de las raíces correcto, el cual su muestra en la Figura 5-44. El programa de MATLAB 5-34 dará también un dibujo del lugar de las raíces correcto. El gráfico que se genera con este programa se muestra en la Figura 5-45.

Programa en MATLAB 5-33

```
num=[1 3.5 1.5];
den=[1 3 2];
v=[-6 6 -6 6];
axis(v);
r=rlocus(num,den);
plot(r,zeros(13,1))
grid
title('Lugar de las raíces de G(s)=K(s+0.5)(s+3)/[(s+1)(s+2)]')
xlabel('Eje Real')
ylabel('Eje Imag')
```

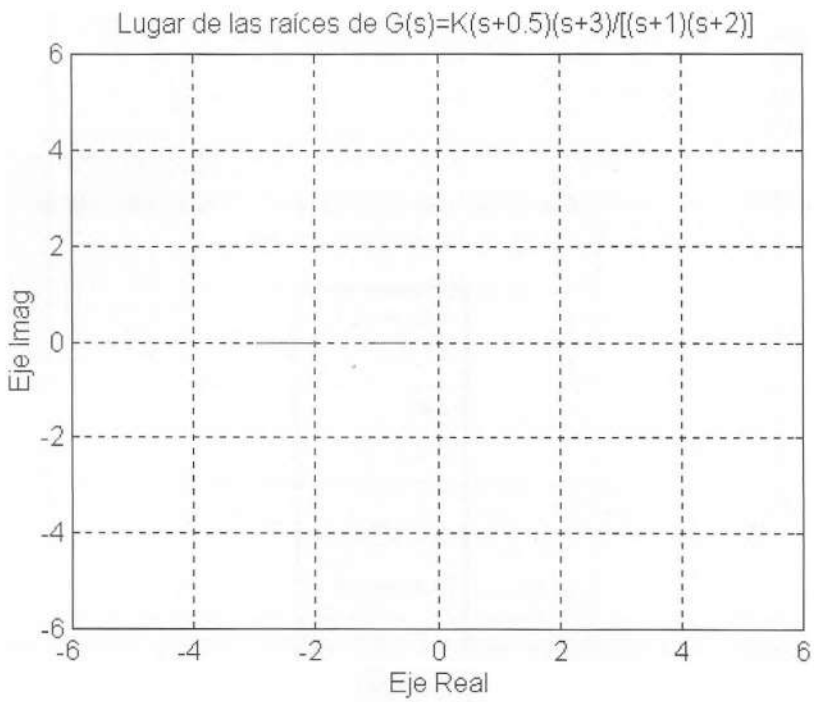


Figura 5-44

Programa en MATLAB 5-34

```
num=[1 3.5 1.5];
den=[1 3 2];
v=[-6 6 -6 6];
axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(s)=K(s+0.5)(s+3)/[(s+1)(s+2)]')
```

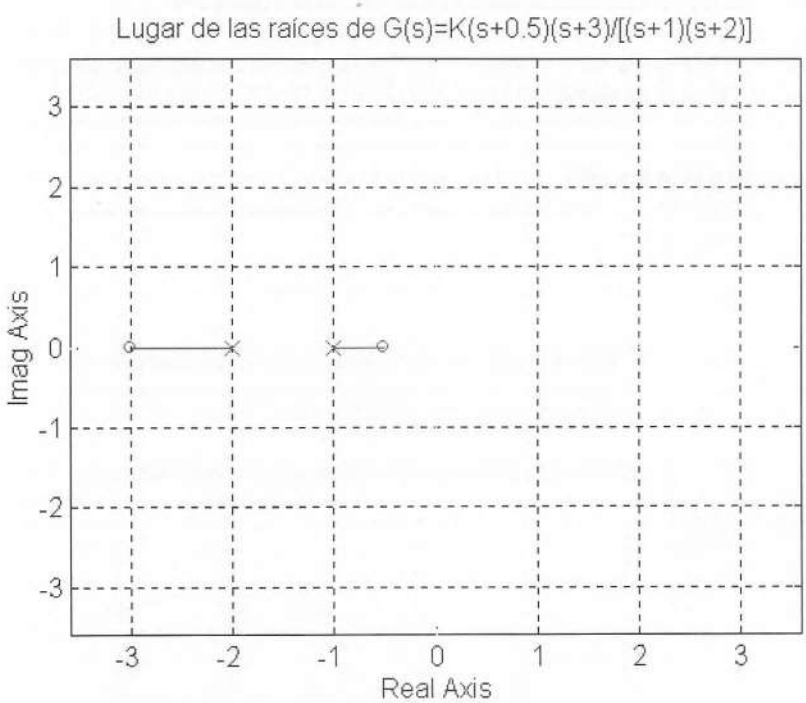


Figura 5-45

EJEMPLO 5-26

Considere el sistema con realimentación positiva que se muestra en la Figura 5-46. Para este sistema con realimentación positiva, la función de transferencia en lazo abierto $G(s)H(s)$ viene dada por

$$G(s)H(s) = -\frac{K(s+0.5)(s+3)}{(s+1)(s+2)}$$

$$= -\frac{K(s^2+3{,}5s+1{,}5)}{s^2+3s+2}$$

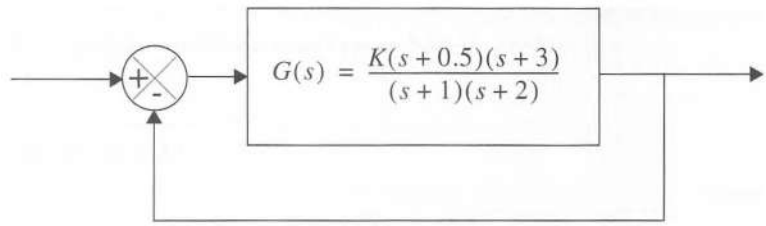


Figura 5-46

Por tanto

$$\text{num} = [-1 \quad -3.5 \quad -1.5]$$
$$\text{den} = [1 \quad 3 \quad 2]$$

Un dibujo del lugar de las raíces para este sistema se puede obtener al introducir el programa de MATLAB 5-35. El lugar de las raíces resultante se muestra en la Figura 5-47.

Para cambiar la región del gráfico, introduzca el programa de MATLAB 5-36. El lugar de las raíces resultante se muestra en la Figura 5-48. (Observe que las flechas se añaden manualmente para indicar la dirección del incremento de K en cada rama).

Programa en MATLAB 5-35

```
num=[-1 -3.5 -1.5];
den=[1 3 2];
rlocus(num,den);
grid
title('Lugar de las raíces de un sistema con realimentación positiva')
```

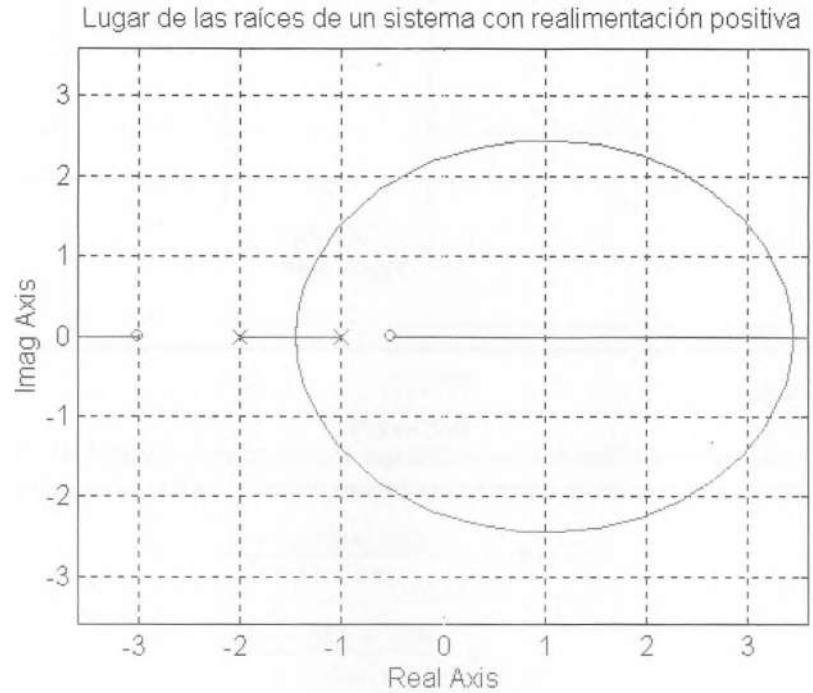


Figura 5-47

Programa en MATLAB 5-36

```
num=[-1 -3.5 -1.5];
den=[1 3 2];
v=[-10 10 -10 10];
axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de un sistema con realimentación positiva')
```

Figura 5-48

El lugar de las raíces tiene un punto de ruptura de salida en $s = -1{,}4495$ y un punto de ruptura de entrada en $s = 3{,}4495$. Cuando el lugar de las raíces se rompe en $s = 3{,}4495$, un camino va directamente a $s = -0{,}5$, que se corresponde con un cero en lazo abierto; el otro camino va a $+\infty$ donde cambia el signo a $-\infty$ y después va a $s = -3$ que se corresponde con el segundo cero del lazo abierto.

Observe que al usar la orden

```
r = rlocus(num,den)
plot(r,'-')
```

puede ocurrir un problema. (Véase el programa de MATLAB 5-37 y su gráfica resultante en la Figura 5-49). En este ejemplo, esta orden da un dibujo del lugar de las raíces erróneo. Todo el eje real se muestra como si fuera una rama del lugar de las raíces. Esto es un error.

Sin embargo, si cambiamos la orden a

```
r = rlocus(num,den)
plot(r,'o')
```

como se da en el programa de MATLAB 5-38, el dibujo del lugar de las raíces es correcto siempre que cuidadosamente se interpreten las ramas del lugar de las raíces sobre el eje real. (Véase la Figura 5-50).

Programa en MATLAB 5-37

```
num=[-1 -3.5 -1.5];
den=[1 3 2];
v=[-10 10 -10 10];
axis(v);
r=rlocus(num,den);
plot(r,'-')
grid
title('Lugar de las raíces erróneo')
xlabel('Eje Real')
ylabel('Eje Imag')
```

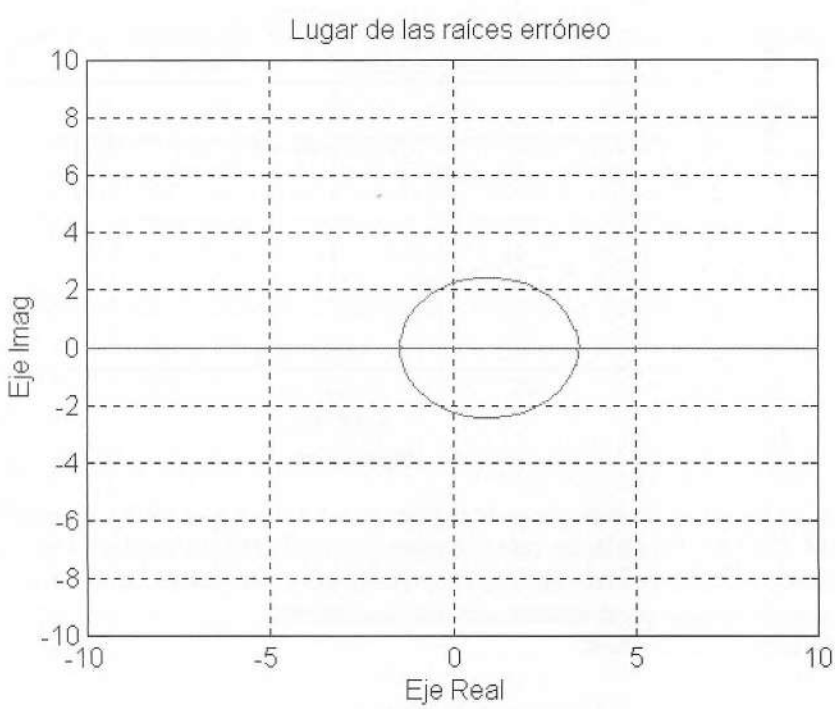


Figura 5-49

Programa en MATLAB 5-38

```
num=[-1 -3.5 -1.5];
den=[1 3 2];
v=[-10 10 -10 10];
axis(v);
r=rlocus(num,den);
plot(r,'o')
grid
title('Lugar de las raíces erróneo')
xlabel('Eje Real')
ylabel('Eje Imag')
```

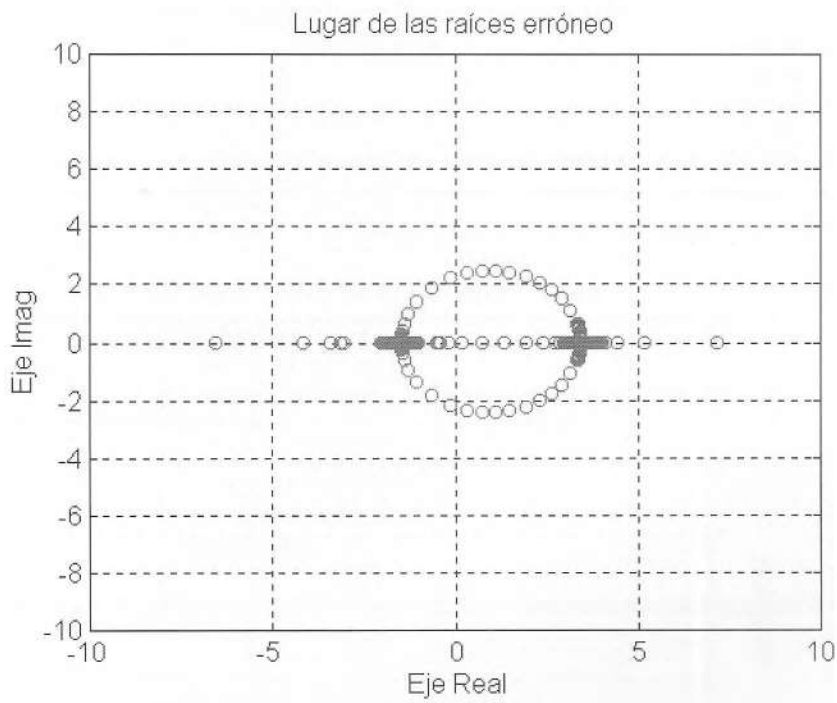


Figura 5-50

5.4 REPRESENTACIÓN DE DOS O MÁS LUGARES DE LAS RAÍCES EN UNA MISMA GRÁFICA

En esta sección se considera el dibujo de dos o más diagramas de lugares de las raíces sobre un único gráfico. Estos casos típicos ocurren cuando se quiere representar el lugar de las raíces y sus asíntotas en un único diagrama.

EJEMPLO 5-27

Considere el sistema cuya función de transferencia en lazo abierto es $G(s)H(s)$ viene dada por

$$G(s)H(s) = \frac{K}{s(s+1)(s+2)}$$

Un dibujo del lugar de las raíces de este sistema se puede obtener introduciendo el programa de MATLAB 5-39 en el computador. La gráfica resultante se muestra en la Figura 5-51.

Programa en MATLAB 5-39

```
num=[0 0 0 1];
den=[1 3 2 0];
rlocus(num,den);
grid
title('Lugar de las raíces de G(s)=K/[s(s+1)(s+2)]')
```

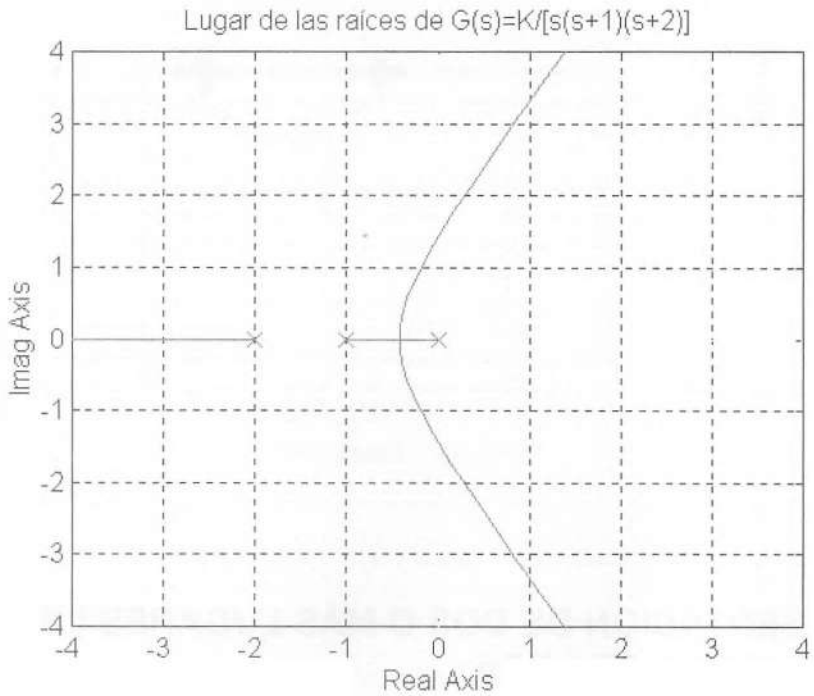


Figura 5-51

A continuación, se van a dibujar el lugar de las raíces y sus asíntotas en un único diagrama. Puesto que la función de transferencia en lazo abierto viene dada por

$$G(s)H(s) = \frac{K}{s(s+1)(s+2)}$$

$$= \frac{K}{s^3 + 3s^2 + 2s}$$

la ecuación para las asíntotas se puede obtener como sigue. Observe que

$$\lim_{s\to\infty} \frac{K}{s^3 + 3s^2 + 2s} \cong \lim_{s\to\infty} \frac{K}{s^3 + 3s^2 + 3s + 1}$$

la ecuación para las asíntotas puede venir dada por

$$G_a(s)H_a(s) = \frac{K}{(s+1)^3}$$

Por lo tanto para el sistema que tenemos

```
num = [0  0  0  1]
den = [1  3  2  0]
```

y para las asíntotas,

```
numa = [0  0  0  1]
dena = [1  3  3  1]
```

Al utilizar las órdenes *rlocus* y *plot*

```
r = rlocus(num,den)
a = rlocus(numa,dena)
plot([r  a])
```

el número de filas de r y de a deben ser iguales. Para asegurar esto, incluimos la constante K en las órdenes. Por ejemplo,

```
K1 = 0:0.1:0.3;
K2 = 0.3:0.005:0.5;
K3 = 0.5:0.5:10;
K4 = 10:5:100;
K = [K1  K2  K3  K4];
r = rlocus(num,den,K)
a = rlocus(numa,dena,K)
y = [r  a]
plot(y,' ')
```

Al incluir la ganancia K en la orden *rlocus* se asegura que la matriz r y la matriz a tienen el mismo número de filas. El programa de MATLAB 5-40 generará un gráfico del lugar de las raíces y de sus asíntotas. Véase la Figura 5-52.

El programa de MATLAB 5-41 utiliza la orden *hold*. El gráfico resultante se muestra en la Figura 5-53.

Programa en MATLAB 5-40

```
num=[0 0 0 1];
den=[1 3 2 0];
numa=[0 0 0 1];
dena=[1 3 3 1];
v=[-4 4 -4 4];
axis(v);
K1=0:0.1:0.3;
K2=0.3:0.005:0.5;
K3=0.5:0.5:10;
K4=10:5:100;
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
a=rlocus(numa,dena,K);
y=[r a];
plot(y,'-g')
grid
title('Lugar de las raíces de G(s)=K/[s(s+1)(s+2)] y asíntotas')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(y,'-g')' visualiza en la pantalla líneas y
%curvas continuas de color verde*****

%*****Dibuje manualmente los polos en lazo abierto sobre la copia impresa*****
```

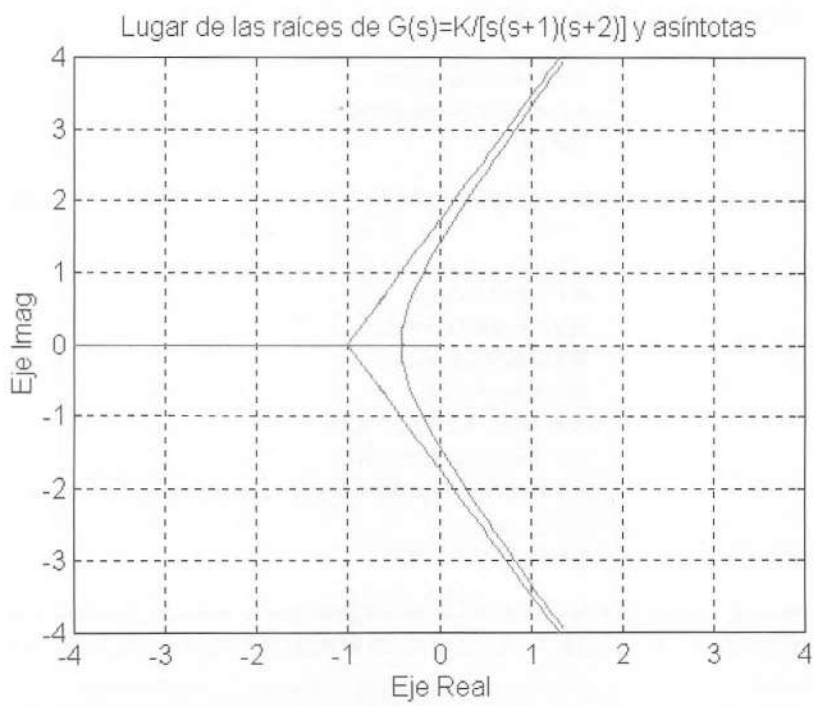


Figura 5-52

EJEMPLO 5-28

Considere un sistema cuya función de transferencia en lazo abierto viene dada por

$$G(s)H(s) = \frac{K(s-0{,}6667)}{s^2(s+1{,}670+j2{,}060)(s+1{,}670-j2{,}060)}$$

$$= \frac{K(s-0{,}6667)}{s^4+3{,}340s^3+7{,}0325s^2}$$

El sistema tiene un cero en lazo abierto en $s = 0{,}6667$ y cuatro polos en lazo abierto en $s = -1{,}670 + j2{,}060$, $s = -1{,}670 - j2{,}060$, $s = 0$ y $s = 0$.

La ecuación para las asíntotas se puede obtener como

$$G_a(s)H_a(s) = \frac{K}{s^3+(3{,}340+0{,}6667)s^2+\cdots}$$

$$\cong \frac{K}{\left(s+\dfrac{3{,}340+0{,}6667}{3}\right)^3}$$

$$= \frac{K}{(s+1{,}3356)^3}$$

$$= \frac{K}{s^3+4{,}0068s^2+5{,}3515s+2{,}3825}$$

Por tanto, introducimos los siguientes numeradores y denominadores en el programa. Para el sistema,

```
num = [0  0  0  1  -0.6667]
den = [1  3.340  7.0325  0  0]
```

Para las asíntotas,

```
numa = [0  0  0  1]
dena = [1  4.0068  5.3515  2.3825]
```

Utilizando la misma aproximación que se discutió en el Ejemplo 5-27, se incluye la ganancia K en la orden *rlocus* como sigue:

```
K1 = 0:1:50
K2 = 50:5:200
K = [K1  K2]
r = rlocus(num,den,K)
a = locus(numa,dena,K)
y = [r  a]
plot(y,' ')
```

Véase el programa de MATLAB 5-42 y la gráfica resultante en la Figura 5-54.

El programa de MATLAB utiliza la orden *hold*. En este caso, no es necesario incluir la ganancia K en la orden *rlocus* ya que los gráficos de r y a se realizan de forma independiente dentro del mismo gráfico. El dibujo utilizando este programa se muestra en la Figura 5-55.

Programa en MATLAB 5-42

```
num=[0 0 0 1 -0.6667];
den=[1 3.34 7.0325 0 0];
numa=[0 0 0 1];
dena=[1 4.0068 5.3515 2.3825];
v=[-4 3 -5 5];
axis(v);
K1=0:0.1:0.3;
K2=0.3:0.005:0.5;
K3=0.5:0.5:10;
K4=10:5:100;
K=[K1 K2 K3 K4];
r=rlocus(num,den,K);
a=rlocus(numa,dena,K);
y=[r a];
plot(y,'-g')
grid
title('Lugar de las raíces de G(s)=K/[s(s+1)(s+2)] y asíntotas')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Observe que la orden 'plot(y,'-g')' visualiza en la pantalla líneas y
%curvas continuas de color verde*****

%*****Dibuje manualmente los polos en lazo abierto sobre la copia impresa*****
```

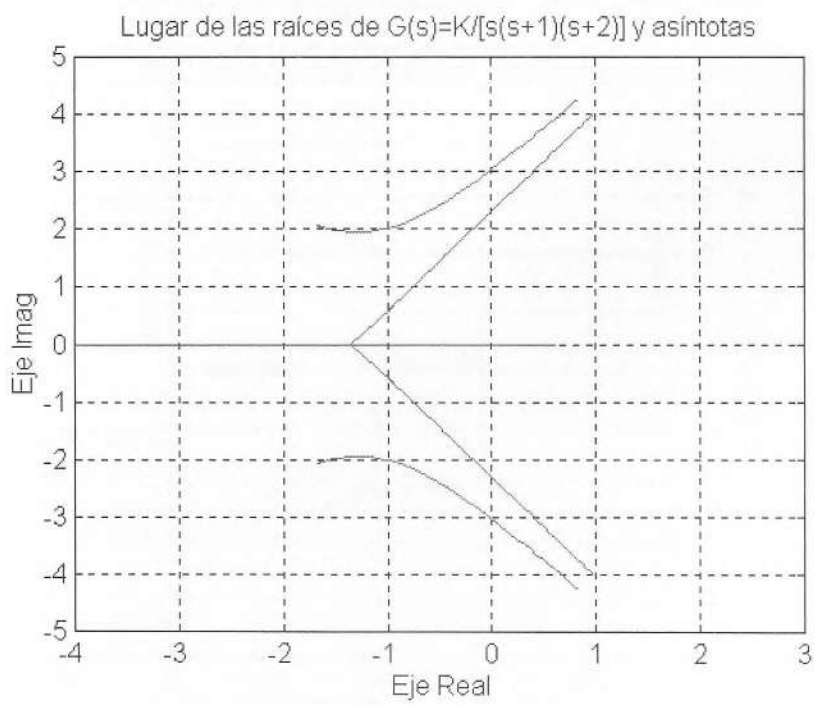


Figura 5-54

Programa en MATLAB 5-43

```
num=[0 0 0 1 -0.6667];
den=[1 3.34 7.0325 0 0];
numa=[0 0 0 1];
dena=[1 4.0068 5.3515 2.3825];
v=[-4 3 -5 5];
axis(v);
r=rlocus(num,den);
plot(r,'o')
hold

Current plot held

a=rlocus(numa,dena,K);
plot(a,'-')
grid
title('Lugar de las raíces de G(s) y asíntotas')
xlabel('Eje Real')
ylabel('Eje Imag')

%*****Dibuje manualmente los polos y ceros en lazo abierto
%sobre la copia impresa*****

%*****Eliminar el mantenimiento de la gráfica*****

hold
Current plot released
```

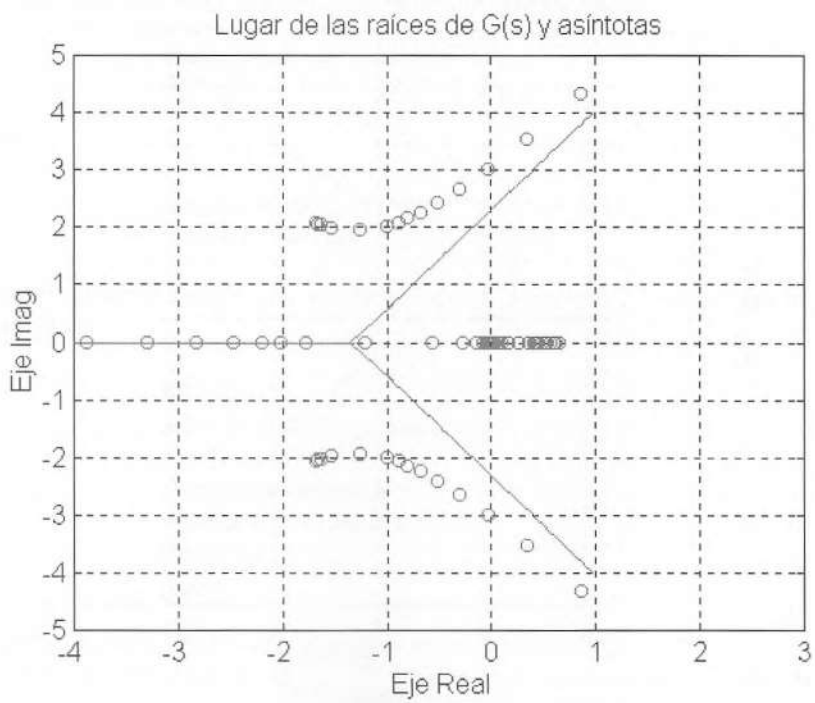


Figura 5-55

5.5 LUGAR DE LAS RAÍCES EN EL PLANO *Z*

La construcción de los dibujos de los lugares de las raíces en el plano z son exactamente iguales a los del plano s. La única diferencia en los dibujos de estos dos planos es la interpretación de la región de estabilidad. En el plano s, los polos en lazo cerrado que se encuentren en el semiplano derecho son polos inestables, mientras que en el plano z los polos en lazo cerrado que se encuentren fuera del círculo unidad centrado en el origen son polos inestables.

EJEMPLO 5-29

Dibuje el lugar de las raíces del sistema mostrado en la Figura 5-56. Suponga que la función de transferencia en lazo abierto $G_D(z)G(z)$ viene dada por

$$G_D(z)G(z) = \frac{0{,}0176K(z + 0{,}8760)}{(z - 0{,}2543)(z - 1)}$$

Para dibujar el lugar de las raíces de este sistema, llamamos K' al producto de la ganancia K por 0.0176, y K' varía desde cero hasta un número grande. Esto significa que $G_D(z)G(z)$ se puede escribir como

$$G_D(z)G(z) = \frac{K'(z + 0{,}8760)}{(z - 0{,}2543)(z - 1)}$$

donde K' es la ganancia total. Al escribir el polinomio denominador no lo multiplicamos por la nueva ganancia K', y el num y el den vienen dados por

```
num = [0   1   0.8760]
den = [1   -1.2543   0.2543]
```

El programa de MATLAB 5-44 producirá el dibujo del lugar de las raíces. Este gráfico se muestra en la Figura 5-57.

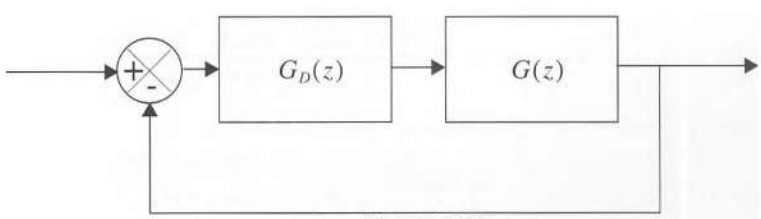


Figura 5-56

Programa en MATLAB 5-44

```
num=[0 1 0.8760]; den=[1 -1.2543 0.2543];
v=[-4 4 -4 4]; axis(v);
rlocus(num,den);

Axis scales auto-ranged
Axis scales frozen

grid, title('Lugar de las raíces de G(z)=K(z+0.8760)/[(z-0.2543)(z-1)]')
```

Figura 5-57

EJEMPLO 5-30

Considere el sistema que se muestra en la Figura 5-58. Dibuje el diagrama del lugar de las raíces. La función de transferencia en lazo abierto viene dada por

$$G_D(z)G(z) = \frac{(1.4 - 1.4z^{-1} + 0.2z^{-2})(0.3679z^{-1} + 0.2642z^{-2})}{(1 - z^{-1})(1 - 0.3679z^{-1})(1 - z^{-1})}$$

$$= \frac{0{,}5151z^3 - 0{,}1452z^2 - 0{,}2963z + 0{,}05284}{z^4 - 2{,}3679z^3 + 1{,}7358z^2 - 0{,}3679z}$$

El programa de MATLAB 5-45 se puede utilizar para generar el lugar de las raíces. El gráfico resultante de este programa se muestra en la Figura 5-59.

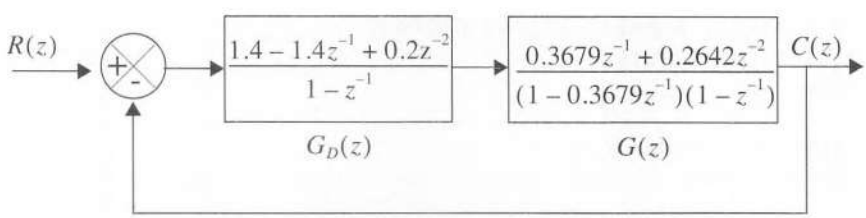


Figura 5-58

Programa en MATLAB 5-45

```
num=[0.5151 -0.1452 -0.2963 0.05284];
den=[1 -2.3679 1.7358 -0.3679 0];
v=[-3 2 -2 2];
axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de un sistema de control digital')
```

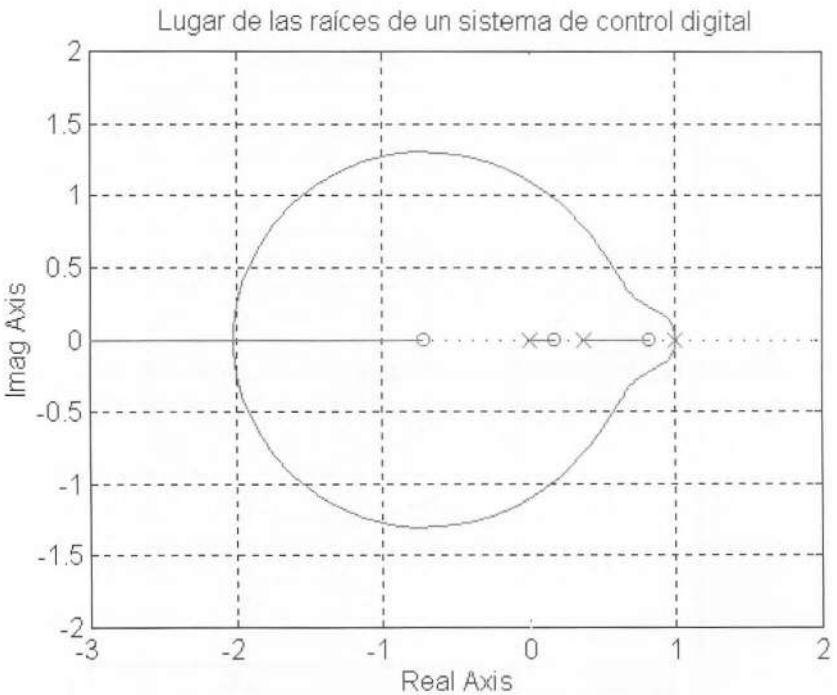


Figura 5-59

EJEMPLO 5-31: CÍRCULO UNIDAD

Al dibujar el diagrama del lugar de las raíces en el plano z, frecuentemente es deseable superponer el diagrama del círculo unidad.

El círculo unidad, centrado en el origen, se puede dibujar fácilmente utilizando la siguiente orden:

```
p = 0:0.01:2*pi;
x = sin(p); y = cos(p);
plot(x,y)
```

Véase el programa de MATLAB 5-46 y el círculo unidad resultante en la Figura 5-60.

Programa en MATLAB 5-46

```
p=0:0.01:2*pi;
x=sin(p);
y=cos(p);
v=[-2 2 -2 2];
axis(v);
plot(x,y)
grid
title('Círculo unidad')
xlabel('Eje Real')
ylabel('Eje Imag')
```

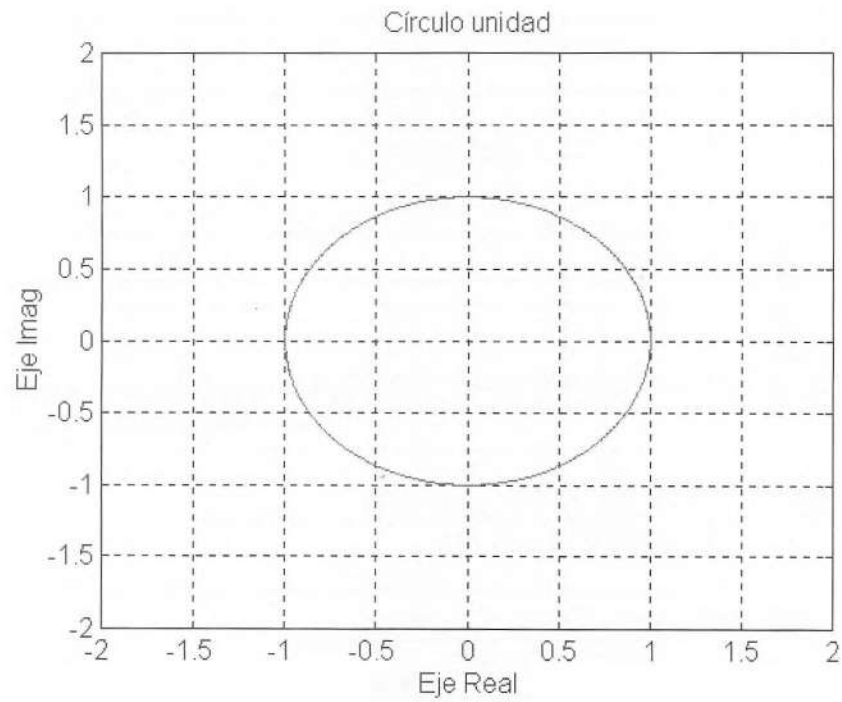


Figura 5-60

EJEMPLO 5-32

Considere el sistema mostrado en la Figura 5-61. Obtenga el lugar de las raíces para el sistema.

Introduzca el programa de MATLAB 5-47 en el computador, el diagrama del lugar de las raíces se muestra en la Figura 5-62.

Suponga que se desea superponer el cículo unidad en el diagrama. Se puede realizar de un forma sencilla utilizando la orden *hold* y la orden para dibujar un círculo unidad, al igual que se presenta en el Ejemplo 5-31. Véase el programa de MATLAB 5-48 y el gráfico resultante en la Figura 5-63.

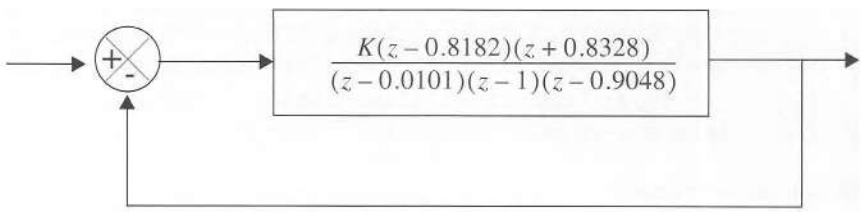


Figura 5-61

Programa en MATLAB 5-47

```
num=[0 1 0.0146 -0.6814];
den=[1 -1.9149 0.9240 -0.0091];
v=[-3 2 -2 2];
axis(v);
rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid
title('Lugar de las raíces de G(z)=K(z-0.8182)(z+0.8328)/[(z-0.0101)(z-0.9048)]')
```

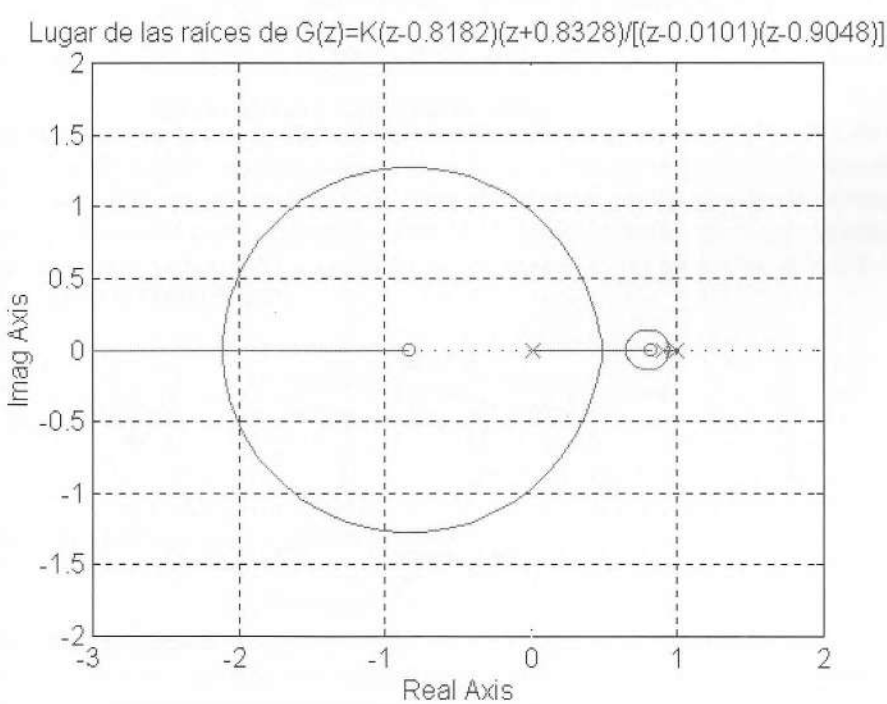


Figura 5-62

Programa en MATLAB 5-48

```
%------Lugar de las raíces y círculo unidad------

num=[0 1 0.0146 -0.6814]; den=[1 -1.9149 0.9240 -0.0091];
v=[-4 2 -2 2]; axis(v); rlocus(num,den);

Axis scales auto-ranged

Axis scales frozen

grid

%*****Para superponer un círculo unidad sobre el lugar de las raíces, introduzca
%la orden hold y una orden para generar un círculo unidad*****

hold

Current plot held

p=0:0.01:2*pi; x=sin(p); y=cos(p); plot(x,y,'--')
title('Lugar de las raíces y círculo unidad')
text(0.8,-0.8, 'Círculo unidad')

%*****Quitar el mantener las gráficas*****

hold

Current plot released
```

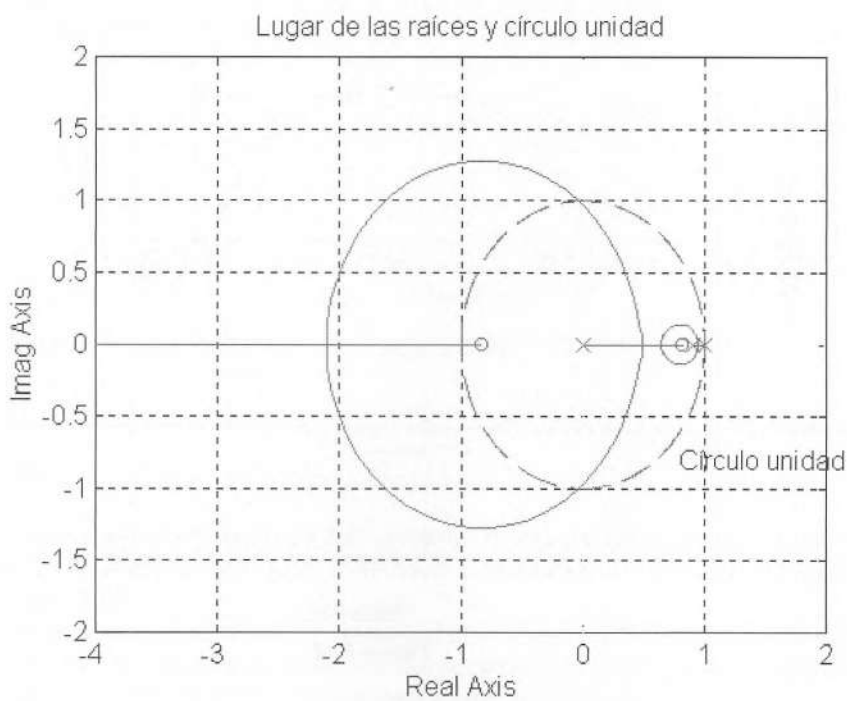


Figura 5-63

EJEMPLO 5-33: SISTEMA DE CONTROL DE RESPUESTA PLANA

Considere el sistema de control mostrado en la Figura 5-64. El periodo de muestreo T es 1 seg. Las funciones de transferencia para el controlador digital y la planta vienen dadas respectivamente por

$$G_D(z) = \frac{2{,}5(1-0{,}6z^{-1})}{1+0{,}75z^{-1}} = \frac{2{,}5(z-0{,}6)}{z+0{,}75}$$

$$G(z) = \frac{(1+z^{-1})z^{-1}}{2(1-z^{-1})^2} = \frac{z+1}{2(z-1)^2}$$

Así,

$$G_D(z)G(z) = \frac{1{,}25(z^2+0{,}4z-0{,}6)}{z^3-1{,}25z^2-0{,}5z+0{,}75}$$

$$= \frac{K(z^2+0{,}4z-0{,}6)}{z^3-1{,}25z^2-0{,}5z+0{,}75}, \qquad K = 1{,}25$$

Este sistema se ha diseñado para mostrar una respuesta plana a entradas rampa. (Las características de la respuesta transitoria de este sistema se examinaron en el Ejemplo 4-9). Obsérvese que un sistema de respuesta plana tiene todos los polos en lazo cerrado en el origen. Para este sistema hay tres polos en lazo cerrado en el origen:

$$\frac{C(z)}{R(z)} = \frac{2{,}5(z^2+0{,}4z-0{,}6)}{(2z+1{,}5)(z^2-2z+1)+2{,}5(z^2+0{,}4z-0{,}}$$

$$= \frac{1{,}25(z^2+0{,}4z-0{,}6)}{z^3}$$

Para obtener el lugar de las raíces del sistema, introduzca el programa de MATLAB 5-49 en el computador. Mire el gráfico del lugar de las raíces que se muestra en la Figura 5-65 y que se ha obtenido con el programa de MATLAB 5-49, se puede observar que cerca del origen el gráfico muestra una forma peculiar. Esto es debido a que el número de puntos calculados cerca del origen es insuficiente. Para examinar las causas de estos lugares de las raíces, utilice el lugar de las raíces con marcas 'o'. El programa de MATLAB 5-50 genera el gráfico mostrado en la Figura 5-66.

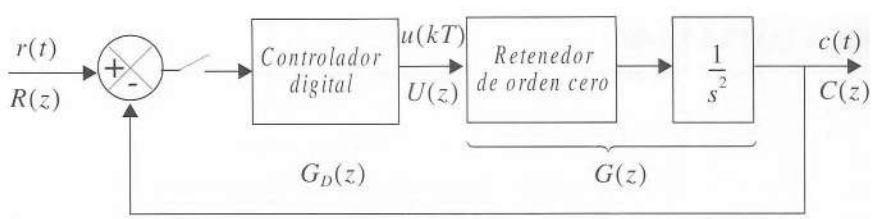


Figura 5-64

Programa en MATLAB 5-49

```
num=[0 1 0.4 -0.6];
den=[1 -1.25 -0.5 0.75];
rlocus(num,den);
grid, title('Sistema de respuesta plana diseñado para entrada en rampa')
```

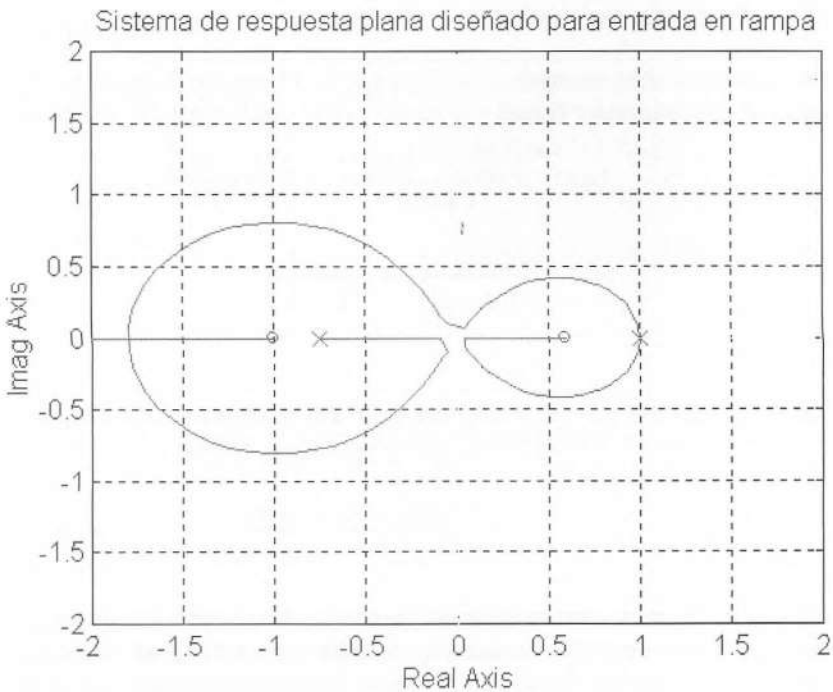


Figura 5-65

En el gráfico de la Figura 5-66, observe que no existen puntos calculados próximos al origen. Para conseguir algunos puntos calculados próximos al origen, especifique los valores de la ganancia K con un espaciado muy pequeño en el área próxima al origen. Por ejemplo, se puede especificar la ganancia K como sigue:

```
K1 = 0:0.01:1.5;
K2 = 1.5:0.1:4.26;
K3 = 4.26:0.001:4.32;
K4 = 4.32:1:10;
K5 = 10:5:100;
K = [K1  K2  K3  K4  K5]
```

Con el programa de MATLAB 5-51 se obtiene el lugar de las raíces de la Figura 5-67.

Programa en MATLAB 5-50

```
num=[0 1 0.4 -0.6];
den=[1 -1.25 -0.5 0.75];
v=[-2 2 -2 2];
axis(v);
r=rlocus(num,den);
grid
plot(r,'o')
title('Sistema de respuesta plana diseñado para entrada en rampa')
xlabel('Eje Real')
ylabel('Eje Imag')
```

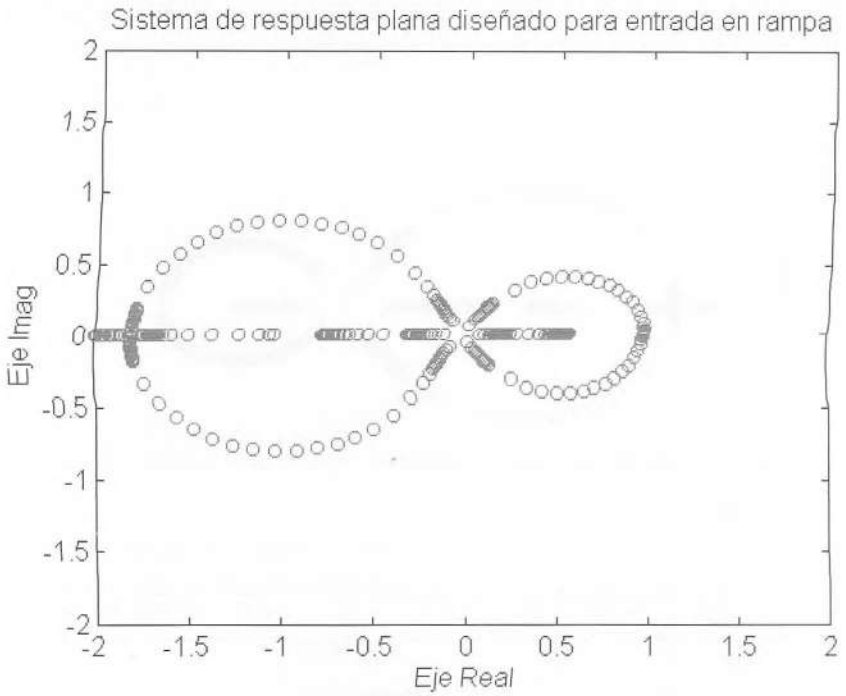


Figura 5-66

Programa en MATLAB 5-51

```
num=[0 1 0.4 -0.6];
den=[1 -1.25 -0.5 0.75];
v=[-2 2 -2 2];
axis(v);
K1=0:0.01:1.5;
K2=1.5:0.1:4.26;
K3=4.26:0.001:4.32;
K4=4.32:1:10;
K5=10:5:100;
K=[K1 K2 K3 K4 K5];
r=rlocus(num,den,K);
grid
plot(r,'o')
title('Sistema de respuesta plana diseñado para entrada en rampa')
xlabel('Eje Real')
ylabel('Eje Imag')
```

Cambie la orden plot en el programa de MATLAB 5-51 de plot(r,'o') a plot(r,'-'), para obtener el lugar de las raíces de la Figura 5-68, el cual está dibujado con líneas y curvas continuas. El programa de MATLAB que genera este dibujo es el programa de MATLAB 5-52.

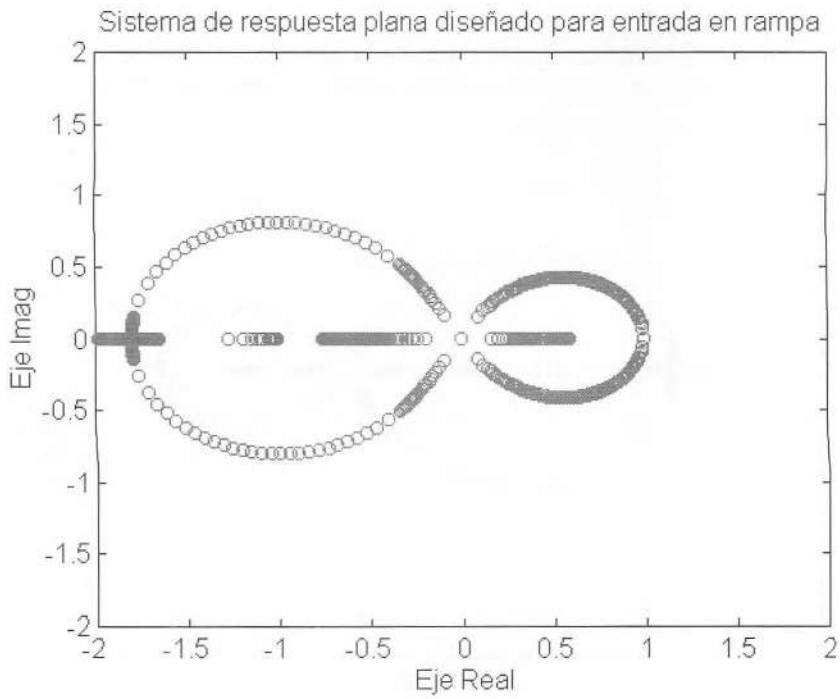


Figura 5-67

Figura 5-68

Programa en MATLAB 5-52

```
num=[0 1 0.4 -0.6];
den=[1 -1.25 -0.5 0.75];
v=[-2 2 -2 2];
axis(v);
K1=0:0.01:1.5;
K2=1.5:0.1:4.26;
K3=4.26:0.001:4.32;
K4=4.32:1:10;
K5=10:5:100;
K=[K1 K2 K3 K4 K5];
r=rlocus(num,den,K);
plot(r,'-')
grid
title('Sistema de respuesta plana diseñado para entrada en rampa')
xlabel('Eje Real')
ylabel('Eje Imag')
text(0,1.3,'Este sistema tiene tres')
text(0,1.1,'polos en lazo cerrado en')
text(0,0.9,'el origen')
```

EJEMPLO 5-34

Considere el sistema de control de respuesta plana mostrado en la Figura 5-69. (La respuesta a escalón y la respuesta a rampa de un sistema de control de respuesta plana se discutieron en el Ejemplo 4-8). La función de transferencia en lazo abierto es

$$G_D(z)G(z) = \frac{0{,}5820(z + 0.7181)}{(z + 0.4180)(z - 1)}$$

Observe que todos los polos del lazo cerrado de los sistemas de control de respuesta plana están localizados en el origen. Para este sistema, hay dos polos en lazo cerrado en el origen:

$$\frac{C(z)}{R(z)} = \frac{0{,}5820(z + 0.7181)}{(z + 0.4180)(z - 1) + 0{,}5820(z + 0{,}7181)}$$

$$= \frac{0{,}5820(z + 0{,}7181)}{z^2}$$

El programa de MATLAB 5-53 genera el lugar de las raíces de la Figura 5-70. Una cuidadosa comprobación revela que la curva cerrada no rodea completamente al origen. Para comprobar el lugar de las raíces próximo al origen, introduzca el programa de MATLAB 5-54 en el computador. El gráfico resultante se muestra en la Figura 5-71.

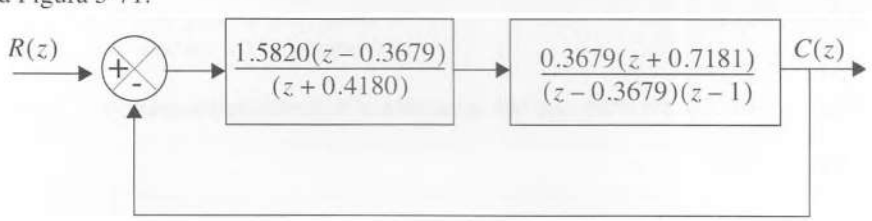


Figura 5-69

El dibujo erróneo del lugar de las raíces próximo al origen y al punto (-1.4,0) se pueden corregir utilizando un paso fijo más pequeño en la ganancia que afecta a esas regiones, como se muestra en el programa de MATLAB 5-55. El gráfico resultante se muestra en la Figura 5-72, el cual parece correcto. Se ha añadido el círculo unidad al diagrama.

Programa en MATLAB 5-53

```
num=[0 1 0.7181];
den=[1 -0.5820 -0.4180];
rlocus(num,den);
grid
title('Lugar de las raíces de un sistema de respuesta plana')
```

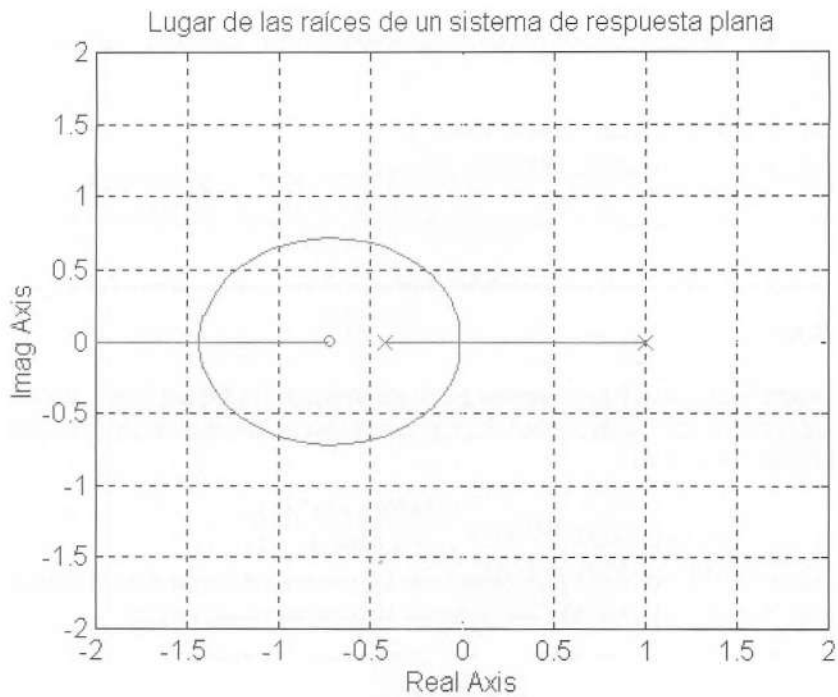


Figura 5-70

Programa en MATLAB 5-54

```
num=[0 1 0.7181];
den=[1 -0.5820 -0.4180];
v=[-2 2 -2 2];
axis(v);
r=rlocus(num,den);
grid
plot(r,'-')
title('Lugar de las raíces de un sistema de respuesta plana')
xlabel('Eje Real')
ylabel('Eje Imag')
```

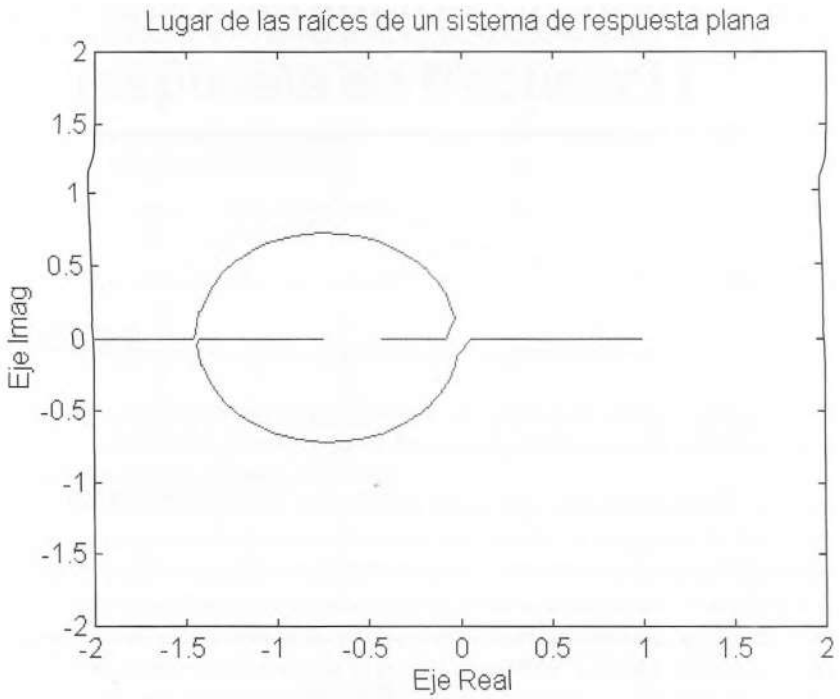


Figura 5-71

Programa en MATLAB 5-55

```
num=[0 1 0.7181]; den=[1 -0.5820 -0.4180];
v=[-2 2 -2 2]; axis(v);
K1=0:0.1:0.5;
K2=0.5:0.001:0.6;
K3=0.6:0.1:3.4;
K4=3.4:0.002:3.6;
K5=3.6:1:30.6;
K=[K1 K2 K3 K4 K5];
r=rlocus(num,den,K);
plot(r,'-')
grid,title('Lugar de las raíces de un sistema de respuesta plana')
xlabel('Eje Real'), ylabel('Eje Imag')
hold

Current plot held

p=0:0.01:2*pi; x=sin(p); y=cos(p); plot(x,y,'--')
text(0.6,-1,'Círculo unidad')
text(0,1.5,'Hay dos polos en lazo')
text(0,1.3,'cerrado en el origen')

%*****Quitar el mantenimimiento de las gráficas*****

hold

Current plot released
```

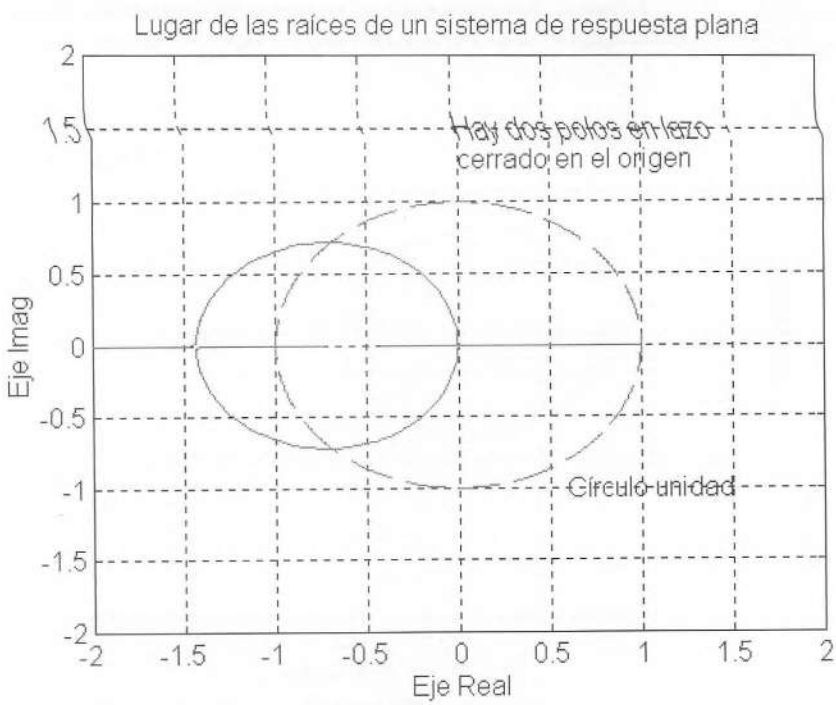

Lugar de las raíces de un sistema de respuesta plana
Hay dos polos en lazo cerrado en el origen
Círculo unidad
Eje Imag
Eje Real
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

Figura 5-72

Capítulo 6 Representación gráfica de la respuesta en frecuencia

6.1 INTRODUCCIÓN

Con el término *respuesta en frecuencia*, se quiere decir la respuesta en estado estacionario de un sistema a una entrada sinusoidal. En los métodos de respuesta en frecuencia, variamos la frecuencia de la señal de entrada en un cierto rango y estudiamos la respuesta resultante.

El criterio de estabilidad de Nyquist nos permite investigar tanto la estabilidad absoluta como la relativa de un sistema lineal en lazo cerrado a partir del conocimiento de su respuesta en frecuencia en lazo abierto. Una ventaja del método de la respuesta en frecuencia es que los tests son en general simples y pueden hacerse de forma precisa mediante el empleo de generadores de señales sinusoidales y de buenos equipos de medida. A menudo las funciones de transferencia de componentes complicados se pueden determinar experimentalmente con los tests de respuesta en frecuencia. Además, el método presenta la ventaja de que se puede diseñar un sistema de manera que sean despreciables los efectos de los ruidos no deseados y que el análisis y el diseño se puede extender a ciertos sistemas de control no lineales.

Aunque la respuesta en frecuencia de un sistema de control muestra una visión cualitativa de la respuesta transitoria, la correlación entre la respuesta en frecuencia y transitoria es indirecta, excepto en el caso de sistemas de segundo orden. Al diseñar un sistema en lazo cerrado ajustamos la característica de respuesta en frecuencia de la función de transferencia en lazo abierto utilizando algunos criterios de diseño con el fin de obtener unas características de respuesta transitoria para el sistema aceptables.

Estructura del capítulo

En la sección 6-1 se han hecho algunos comentarios de caracter introductorio. Las secciones 6-2 a 6-5 tratan sobre los sistemas de control en tiempo continuo y la sección 6-6 presenta el caso de los sistemas en tiempo discreto. Específicamente, la sección 6-2 muestra algunos conceptos sobre los métodos de respuesta en frecuencia. La sección 6-3 analiza la representación de los diagramas de Bode con MATLAB. En primer lugar se estudian los diagramas de Bode de sistemas representados como funciones de transferencia y a continuación la de sistemas definidos en el espacio de estados. La sección 6-4 se dedica a la representación de Nyquist. Presentamos los diagramas de Nyquist de funciones de transferencia y de sistemas definidos en el espacio de estados. La sección 6-5 presenta algunos ejemplos que realizan el diseño de sistemas de control utilizando los diagramas de Bode. Finalmente, la sección 6-6 trata la respuesta en frecuencia de sistemas de control en tiempo discreto en el plano w.

6.2 FUNDAMENTOS BÁSICOS

Esta sección presenta conceptos básicos que son necesarios para comprender la respuesta en frecuencia, los diagramas de Bode, los diagramas de Nyquist, el criterio de estabilidad de Nyquist y cosas semejantes. En esta sección estamos interesados en los sistemas de tiempo continuo. En la sección 6-6 se presenta material específico relativo a los sistemas de tiempo discreto.

Salida en estado estacionario a una entrada sinusoidal

Sea el sistema lineal e invariante en el tiempo que se muestra en la Figura 6-1. Para este sistema

$$\frac{Y(s)}{X(s)} = G(s)$$

La entrada $x(t)$ es sinusoidal y viene dada por:

$$x(t) = X \operatorname{sen} \omega t$$

Se puede demostrar fácilmente que si el sistema es estable, entonces la salida $y(t)$ se puede expresar como

$$y(t) = Y \operatorname{sen}(\omega t + \phi)$$

donde

$$Y = X|G(j\omega)|$$

y

$$\phi = \angle G(j\omega) = \tan^{-1}\left[\frac{\text{parte imaginaria de } G(j\omega)}{\text{parte real de } G(j\omega)}\right]$$

Un sistema lineal e invariante en el tiempo que está sujeto a una entrada sinusoidal en estado estacionario tendrá una salida también sinusoidal de la misma frecuencia que la de la entrada. Sin embargo la amplitud y la fase de la salida serán en general diferentes de las de la entrada. De hecho, la amplitud de la salida viene dada por el producto de la amplitud de la entrada y $|G(jw)|$, mientras que la fase difiere de la de la entrada en la cantidad $\phi = \angle G(jw)$. En la Figura 6-2 se muestra un ejemplo de señales de entrada y salida sinusoidales.

Observe que para entradas sinusoidales

$$|G(j\omega)| = \left|\frac{Y(j\omega)}{X(j\omega)}\right| = \text{razón entre las amplitudes de la sinusoide de salida y la sinusoide de entrada}$$

$$\angle G(j\omega) = \angle\frac{Y(j\omega)}{X(j\omega)} = \text{desfase entre las sinusoides de salida y de entrada}$$

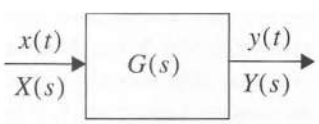


Figura 6-1

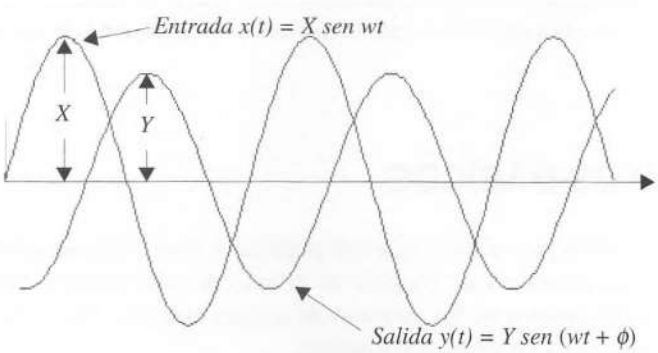


Figura 6-2

Por tanto, las características de respuesta de un sistema a una entrada sinusoidal se pueden obtener directamente a partir de

$$\frac{Y(j\omega)}{X(j\omega)} = G(j\omega)$$

La función de transferencia sinusoidal $G(jw)$ que es la razón de $Y(jw)$ a $X(jw)$ es una cantidad compleja que puede representarse por su magnitud y el ángulo de fase con la frecuencia como parámetro. (Un ángulo de fase negativo se llama un *retardo de fase* y un angulo de fase positivo es un *adelanto de fase*). La función de transferencia sinusoidal de cualquier sistema lineal se obtiene sustituyendo jw por s en la función de transferencia del sistema.

Presentación de características de respuesta en frecuencia en formas gráficas

La función de transferencia sinusoidal, una función compleja de la frecuencia w, se caracteriza por su magnitud y ángulo de fase, con la frecuencia como parámetro. Hay dos representaciones comúnmente utilizadas para las funciones de transferencias sinusoidales:

1. Diagrama de Bode o representación logarítmica.
2. Diagrama de Nyquist o representación polar.

En este capítulo presentaremos detalladamente los diagramas de Bode y de Nyquiist. (Hay otra forma de representar la función de transferencia sinusoidal. Este diagrama representa a la magnitud en logaritmo respecto de la fase en grados. Sin embargo MATLAB trata con los diagramas de Bode y de Nyquist y no con este otro tipo de diagrama y por tanto no los estudiaremos en este capítulo).

Diagramas de Bode o diagramas logarítmicos

Una función de transferencia sinusoidal se puede representar en dos diagramas separados, uno que da la magnitud respecto de la frecuencia y otro que da el ángulo de fase también con la frecuencia. Un diagrama de Bode o diagrama logarítmico consiste en dos gráficas. La primera es una gráfica del logaritmo de la magnitud de una función de transferencia sinusoidal; la segunda es una gráfica del ángulo de fase. Ambas se representan contra la frecuencia en escala logarítmica.

La representación estándar de la magnitud logarítmica de G(jw) es $20 \log|G(jw)|$, donde la base del logaritmo es 10. La unidad utilizada en esta representación de la magnitud es el decibelio, en forma abreviada dB.

Observe que un número mayor que la unidad tiene un valor positivo en decibelios, mientras que un número más pequeño que la unidad tiene un valor negativo. Cuando un número aumenta por un factor de 10, el correspondiente valor en decibelios aumenta en un factor de 20. Esto se puede ver a partir de la siguiente relación:

$$20 \log(K \times 10) = 20 \log K + 20$$

Note también que, cuando se expresa en decibelios, el inverso de un número difiere de su valor solamente en signo; esto es, para el número K

$$20 \log K = -20 \log \frac{1}{K}$$

En la representación logarítmica, las curvas se dibujan en papel semilogarítmico, utilizando la escala logarítmica para la frecuencia y la escala lineal o bien para la magnitud (en decibelios) o el ángulo de fase (en grados). El rango de frecuencia de interés determina el número de ciclos logarítmicos que se necesitan sobre el eje de abcisas.

En los diagramas de Bode, las relaciones de frecuencia se expresan en términos de octavas o décadas. Una octava es una banda de frecuencia que va desde ω_1 hasta $2\omega_1$, donde ω_1 es cualquier valor de frecuencia. Una década es una banda de frecuencia que va desde ω_1 hasta $10\omega_1$, donde otra vez ω_1 es cualquier frecuencia. (En

la escala logarítmica del papel semilogarítmico, cualquier relación de frecuencia dada se puede representar por la misma distancia horizontal. Por ejemplo, la distancia horizontal desde $\omega = 1$ a $\omega = 10$ es igual a la que hay desde $\omega = 3$ hasta $\omega = 30$).

El diagrama de Bode es útil porque muestra ambas características de baja y alta frecuencia de la función de transferencia. La expansión del rango de baja frecuencia mediante la utilización de una escala logarítmica para la frecuencia es muy ventajosa puesto que las características en baja frecuencia son más importantes en los sistemas prácticos. (Observe que debido a la escala de frecuencias logarítmicas, es imposible representar laa curvas hasta la frecuencia cero; sin embargo, esto no plantea ningún problema serio).

Factores integral y derivativo

La magnitud logarítmica de $1/j\omega$ en decibelios es

$$20 \log\left|\frac{1}{j\omega}\right| = -20 \log\omega \text{ dB}$$

El ángulo de fase de $1/j\omega$ es constante e igual a -90°.

Si el logaritmo de la magnitud -20 log ω dB se representa respecto de ω en una escala logarítmica, es una línea recta. Como

$$(-20 \log 10\omega) \text{ dB} = (-20 \log\omega - 20) \text{ dB}$$

la pendiente de la línea recta es -20 dB/década.

Análogamente, el logaritmo de la magnitud de $j\omega$ en decibelios es

$$20 \log|j\omega| = 20 \log\omega \text{ dB}$$

El ángulo de fase de $j\omega$ es constante e igual a 90°. La curva del logaritmo de la magnitud es una línea recta con una pendiente de 20 dB/década. Las Figuras 6-3 y 6-4 muestran respectivamente los diagramas de Bode de $1/j\omega$ y $j\omega$.

Factor de primer orden

El logaritmo de la magnitud del factor de primer orden $1/(1 + j\omega T)$ es

$$20 \log\left|\frac{1}{1 + j\omega T}\right| = -20 \log\sqrt{1 + \omega^2 T^2} \text{ dB}$$

Para bajas frecuencias, tales que $\omega << 1/T$, el logaritmo de la magnitud se puede aproximar por

$$-20 \log\sqrt{1 + \omega^2 T^2} = -20 \log 1 = 0 \text{ dB}$$

Así, la curva del logaritmo de la magnitud a bajas frecuencias es la línea constante de 0-dB. Para altas frecuencias, tales que $\omega >> 1/T$,

$$-20 \log\sqrt{1 + \omega^2 T^2} \cong -20 \log\omega T \text{ dB}$$

*Esto es una expresión aproximada para el rango de altas frecuencias. En $\omega = 1/T$, el logaritmo de la mag*nitud es igual a 0 decibelios y en $\omega = 10/T$ es -20 dB. Por tanto el valor de -20 log ωT disminuye en 20 decibelios en cada década de ω. Para $\omega >> 1/T$, la curva del logaritmo de la magnitud es una línea recta con una pendiente de -20 dB/década. Nuestro análisis muestra que el diagrama de Bode o representación logarítmica de la curva de la respuesta en frecuencia del factor $1/(1 + j\omega T)$ se puede aproximar por dos líneas rectas asíntotas, una línea recta en 0 dB para el rango de frecuencias $0 < \omega < 1/T$ y la otra línea recta con pendiente -20 dB/década para el rango de frecuencias $1/T < \omega < \infty$. La curva exacta del logaritmo de la magnitud, las asíntotas y la curva exacta del ángulo de fase del factor $1/(1 + j\omega T)$ se muestran en la Figura 6-5.

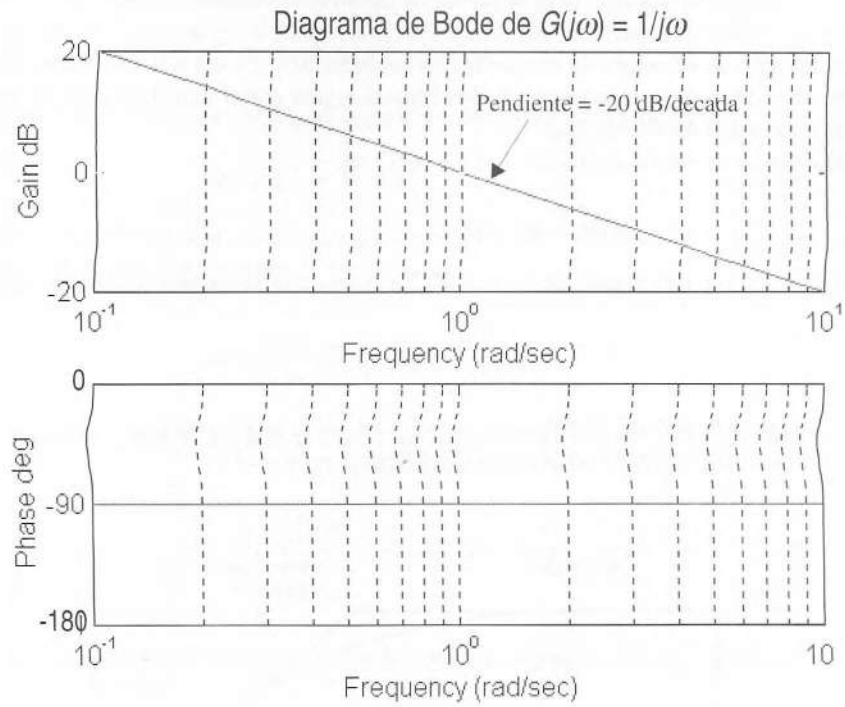


Figura 6-3

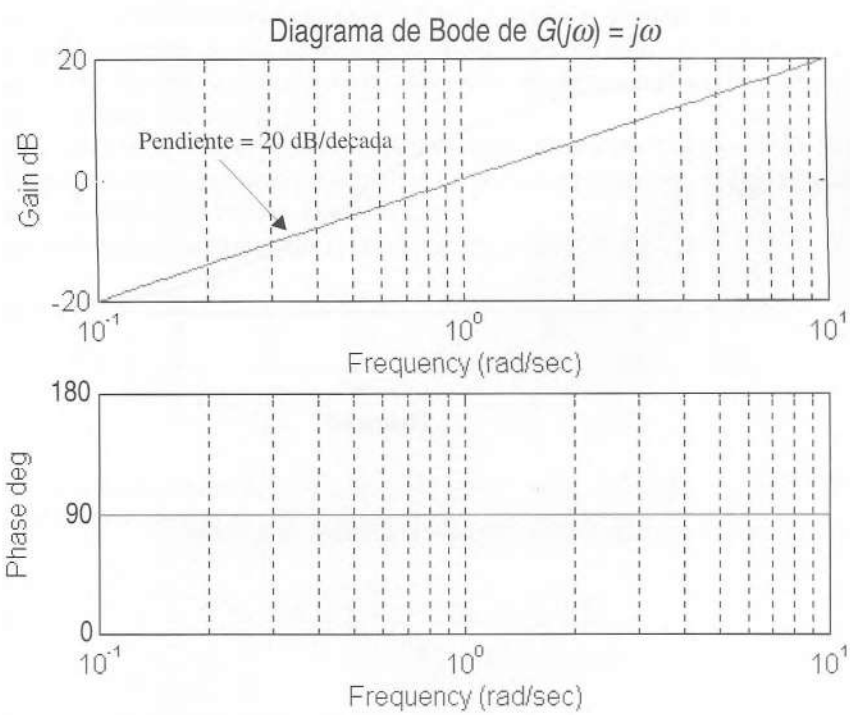


Figura 6-4

La frecuencia a la cuál las dos asíntotas se juntan se llama *frecuencia esquina* o *frecuencia de corte*. Para el factor $1/(1+j\omega T)$, la frecuencia $\omega = 1/T$ es la frecuencia esquina, ya que en $\omega = 1/T$ las dos asíntotas tienen el mismo valor.(La expresión asintótica a baja frecuencia en $\omega = 1/T$ es 20 log 1 dB = 0 dB, y la expresión asintótica a alta frecuencia en $\omega = 1/T$ es también 20 log 1 dB = 0 dB). La frecuencia esquina divide a la curva de la respuesta en frecuencia en dos regiones, una curva para la región de baja frecuencia y una curva para la región de alta frecuencia.

El ángulo exacto de fase ϕ del factor $1/(1+j\omega T)$ es

$$\phi = -\tan^{-1}\omega T$$

En la frecuencia cero, el ángulo de fase es 0°. En la frecuencia esquina, el ángulo de fase es

$$\phi = -\tan^{-1}\frac{T}{T} = -\tan^{-1}1 = -45^\circ$$

En el infinito, el ángulo de fase llega a ser -90°. Al ser el ángulo de fase una función tangente inversa, el ángulo de fase es simétrico respecto al punto de inflexión en ϕ = -45°.

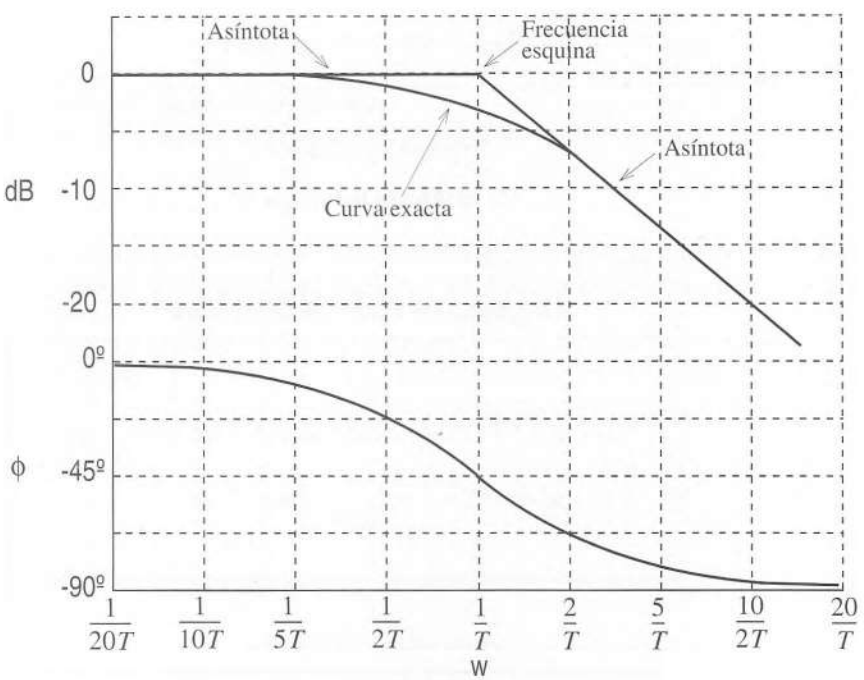


Figura 6-5

Factor cuadrático

Los sistemas de control a menudo tienen factores cuadráticos de la forma

$$\frac{1}{1+2\zeta\left(j\frac{\omega}{\omega_n}\right)+\left(j\frac{\omega}{\omega_n}\right)^2} \qquad (6\text{-}1)$$

Si $\zeta > 1$, este factor cuadrático se puede expresar como un producto de dos factores de primer orden con polos reales. Si $0 < \zeta < 1$, este factor cuadrático es el producto de dos factores complejos conjugados.

La curva de la respuesta en frecuencia asintótica se puede obtener como sigue. Puesto que

$$20 \log\left|\frac{1}{1+2\zeta\left(j\frac{\omega}{\omega_n}\right)+\left(j\frac{\omega}{\omega_n}\right)^2}\right| = -20 \log\sqrt{\left(1-\frac{\omega^2}{\omega_n^2}\right)^2+\left(2\zeta\frac{\omega}{\omega_n}\right)^2}$$

para las bajas frecuencias ya que $\omega << \omega_n$, el logaritmo de la magnitud llega a ser

$$-20 \log 1 = 0 \text{ dB}$$

La asíntota a bajas frecuencias es una línea horizontal sobre 0 dB. Para altas frecuencias tal que $\omega >> \omega_n$, el logaritmo de la magnitud llega a ser

$$-20 \log\frac{\omega^2}{\omega_n^2} = -40\log\frac{\omega}{\omega_n} \text{ dB}$$

La ecuación para la asíntota de altas frecuencias es una línea recta con una pendiente de –40 dB/década, puesto que

$$-40 \log\frac{10\omega}{\omega_n} = -40 - 40 \log\frac{\omega}{\omega_n}$$

La asíntota de alta frecuencia intersecta a la de baja frecuencia en $\omega = \omega_n$, ya que en esta frecuencia

$$-40 \log\frac{\omega}{\omega_n} = -40 \log 1 = 0 \text{ dB}$$

Esta frecuencia es la frecuencia esquina para el factor cuadrático considerado.

Las dos asíntotas obtenidas de esta forma son independientes del valor de ζ. Próximo a la frecuencia $\omega = \omega_n$, aparece un pico de resonancia como se puede esperar de la expresión (6-1). El coeficiente de amortiguamiento ζ determina la magnitud de este pico.

La Figura 6-6 muestra un diagrama de Bode para el factor cuadrático dado por (6-1). Se muestran las curvas exactas del logaritmo de la magnitud junto con las líneas rectas asíntotas y las curvas exactas del ángulo de fase para el factor cuadrático con varios valores de ζ.

El ángulo de fase del factor cuadrático $[1 + 2\zeta(j\omega/\omega_n) + (j\omega/\omega_n)^2]^{-1}$ es

$$\phi = \angle\frac{1}{1+2\zeta\left(j\frac{\omega}{\omega_n}\right)+\left(j\frac{\omega}{\omega_n}\right)^2} = -\tan^{-1}\left[\frac{2\zeta\frac{\omega}{\omega_n}}{1-\left(\frac{\omega}{\omega_n}\right)^2}\right]$$

El ángulo de fase es una función de ω y de ζ. En $\omega = 0$, el ángulo de fase es igual a 0°. En la frecuencia esquina, $\omega = \omega_n$, el ángulo de fase es -90° independientemente de ζ ya que

$$\phi = -\tan^{-1}\left(\frac{2\zeta}{0}\right) = -\tan^{-1}\infty = -90°$$

En $\omega = \infty$, el ángulo de fase llega a ser de -180°. La curva del ángulo de fase es simétrica respecto del punto de inflexión, el punto donde ϕ = -90°.

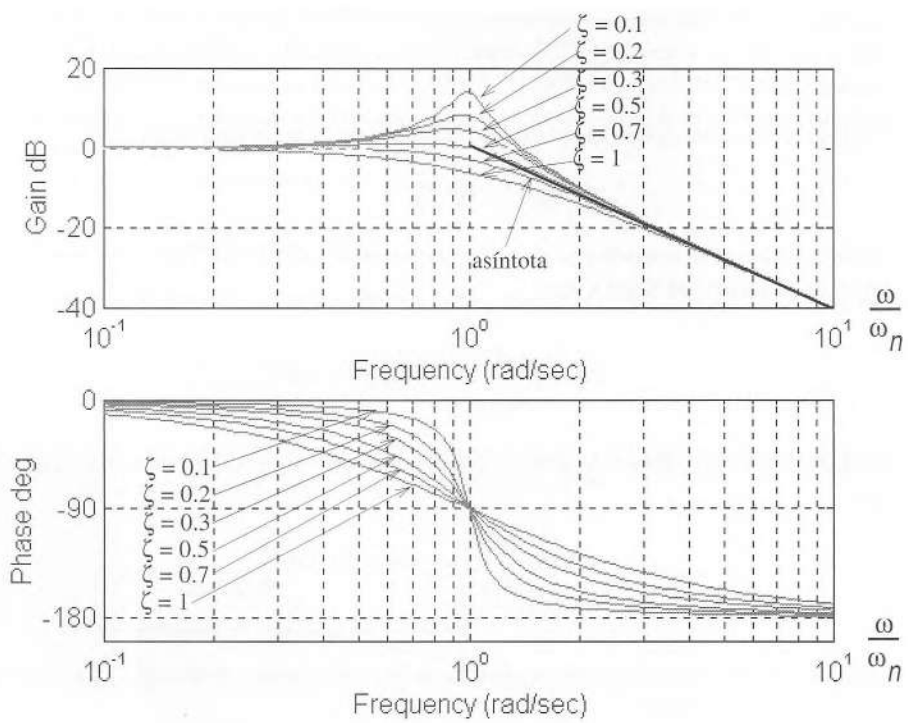


Figura 6-6

Relación entre tipo de sistema y los diagramas de magnitud-fase

Considere el sistema mostrado en la Figura 6-7. La constante de error de posición estática K_p, la constante de error de velocidad estática K_v y la constante de error de aceleración estática K_a se definen como sigue:

$$K_p = \lim_{s \to 0} G(s)H(s) = G(0)H(0) \tag{6-2}$$

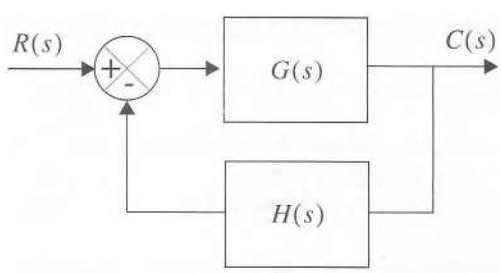


Figura 6-7

$$K_v = \lim_{s \to 0} sG(s)H(s) \tag{6-3}$$

$$K_a = \lim_{s \to 0} s^2 G(s)H(s) \tag{6-4}$$

Estas constantes de error estáticas describen el comportamiento a bajas frecuencias de los sistemas de tipo 0, tipo 1 y tipo 2. Recuerde que el número de integradores ($1/s$) encontrados en $G(s)H(s)$ determinan el tipo del sistema; es decir, si $G(s)H(s)$ tiene 0, 1, 2, ... integradores, al sistema se le denominará de tipo 0, de tipo 1, de

tipo 2, ..., respectivamente. Observe que, para un sistema dado, únicamente una de las constantes de error estática es finita y significativa. (Cuanto mayor sea el valor de la constante de error estática finita más grande es la ganancia del lazo cuando ω tiende a cero).

El tipo del sistema determina la pendiente de la curva del logaritmo de la magnitud a bajas frecuencias. De este modo la información sobre la existencia y la magnitud del error en estado estacionario de un sistema de control a una entrada dada se puede determinar observando la región de bajas frecuencias de la curva del logaritmo de la magnitud.

Determinación de la constante de error de posición estática

La Figura 6-8 muestra un ejemplo de la representación de la curva del logaritmo de la magnitud de un sistema de tipo 0. En tal sistema, la magnitud de $G(j\omega)H(j\omega)$ es igual a K_p a bajas frecuencias, o

$$\lim_{\omega \to 0} G(j\omega)H(j\omega) = K_p$$

Se sigue que la asíntota a bajas frecuencias es una línea horizontal en 20 log K_p dB.

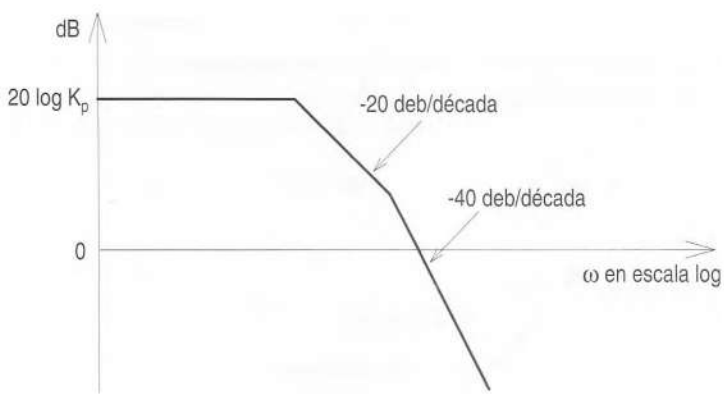


Figura 6-8

Determinación de la constante de error de velocidad estática

La Figura 6-9 muestra un ejemplo de la representación de la curva del logaritmo de la magnitud de un sistema de tipo 1. La intersección del segmento inicial de -20 dB/década (o su prolongación) con la línea $\omega = 1$ tiene la magnitud 20 log K_v. Esto se puede ver como sigue: En un sistema de tipo 1,

$$G(j\omega)H(j\omega) = \frac{K_v}{j\omega}, \qquad \text{para } \omega << 1 \tag{6-5}$$

De este modo

$$20 \log \left| \frac{K_v}{j\omega} \right|_{\omega = 1} = 20 \log K_v$$

La intersección del segmento inicial de -20 dB/década (o su prolongación) con la línea de 0 dB tiene una frecuencia numéricamente igual a K_v. Para ver esto, defina la frecuencia en esta intersección como ω_1; a partir de (6-5) se tiene que

$$\left| \frac{K_v}{j\omega_1} \right| = 1$$

o

$$K_v = \omega_1$$

Determinación de la constante de error de aceleración estática

La Figura 6-10 muestra un ejemplo de la representación de la curva del logaritmo de la magnitud de un sistema de tipo 2. La intersección del segmento inicial de -40 dB/década (o su prolongación) con la línea $\omega = 1$ tiene la magnitud 20 log K_a, ya que a bajas frecuencias

$$G(j\omega)H(j\omega) = \frac{K_a}{(j\omega)^2}, \qquad \text{para } \omega << 1 \tag{6-6}$$

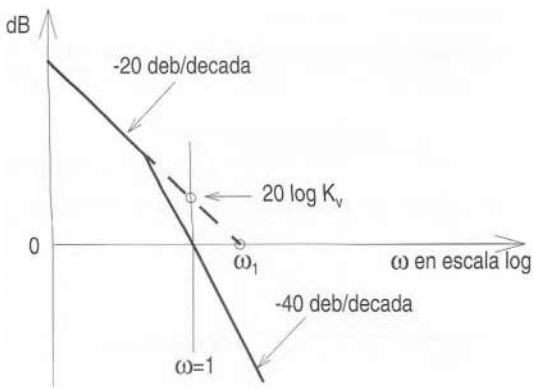


Figura 6-9

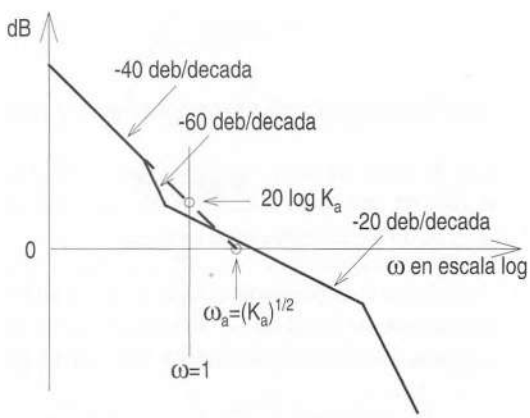


Figura 6-10

se sigue que

$$20 \log \left| \frac{K_a}{(j\omega)^2} \right|_{\omega \to 1} = 20 \log K_a$$

La frecuencia ω_a en la intersección del segmento inicial de -40 db/década (o su prolongación) con la línea de 0 dB da el valor numérico de la raiz cuadrada de K_a. Esto se puede ver como sigue. Defina la frecuencia en esta intersección como ω_a. A partir de (6-6) tenemos

$$\left| \frac{K_a}{(j\omega)^2} \right| = 1$$

o

$$\omega_a = \sqrt{K_a}$$

Sistemas de fase mínima y sistemas de fase no mínima

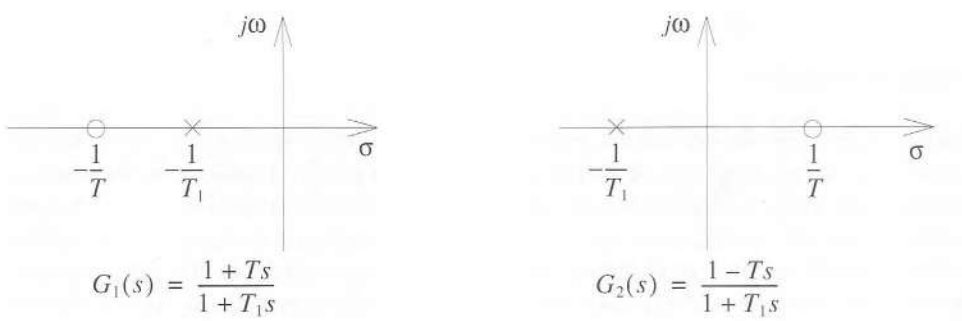
Las funciones de transferencia que no tienen polos ni ceros en el semiplano derecho del plano s son funciones *de transferencia de fase mínima, mientras que si tienen polos y/o ceros en el semiplano derecho del plano s son* funciones de transferencia de fase no mínima. Los sistemas con funciones de transferencia de fase mínima se llaman sistemas de fase mínima; y los que tienen funciones de transferencia no mínimas se denominan sistemas de fase no mínima.

Para sistemas con la misma característica de magnitud, la variación en el ángulo de fase de funciones de transferencia de fase mínima es mínima para todos estos sistemas, mientras que la variación en el ángulo de fase de cualquier función de transferencia de fase no mínima es mayor que este mínimo.

Considere como ejemplo los dos sistemas cuyas funciones de transferencia senoidales son respectivamente,

$$G_1(j\omega) = \frac{1+j\omega T}{1+j\omega T_1}, \qquad G_2(j\omega) = \frac{1-j\omega T}{1+j\omega T_1}, \qquad 0 < T < T_1$$

Las configuraciones de ceros y polos de estos sistemas se muestran en la Figura 6-11. Las dos funciones de transferencia senoidales tienen la misma característica de magnitud, pero tienen distintas características de ángulo de fase como se muestra en la Figura 6-12.



$$G_1(s) = \frac{1+Ts}{1+T_1 s} \qquad G_2(s) = \frac{1-Ts}{1+T_1 s}$$

Figura 6-11

Estos dos sistemas se diferencian entre sí por el factor

$$G(j\omega) = \frac{1-j\omega T}{1+j\omega T}$$

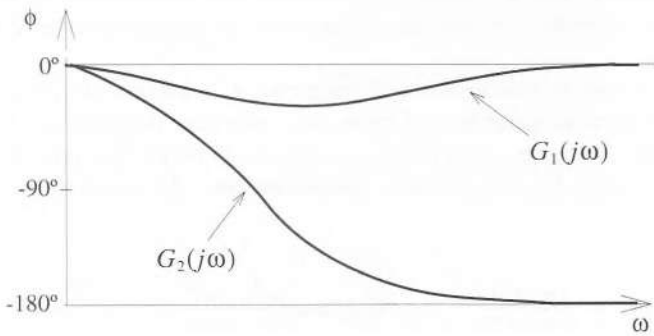
La magnitud del factor $(1-j\omega T)/(1+j\omega T)$ es siempre la unidad. Mientras que el ángulo de fase es igual a $-2\tan^{-1}\omega T$ y varía de 0 a -180° cuando ω se incrementa de cero a infinito.

Para un sistema de fase mínima, las características de magnitud y de ángulo de fase están directamente relacionadas. Esto significa que si la curva de magnitud de un sistema se especifica sobre una variación de frecuencia de cero a infinito, la curva de ángulo de fase está determinada unívocamente, y viceversa. Esto, sin embargo, no se mantiene para sistemas de fase no mínima.

Las situaciones de fase no mínima se pueden presentar de dos formas diferentes. Una es simplemente cuando un sistema incluye un elemento o varios elementos de fase no mínima. La otra situación se puede presentar cuando un lazo secundario es inestable.

Para un sistema de fase mínima, el ángulo de fase en $\omega = \infty$ llega a ser $-90°(q - p)$, donde p y q son respectivamente los grados de los polinomios del numerador y del denominador de la función de transferencia. Para un sistema de fase no mínima, el ángulo de fase en $\omega = \infty$ difiere de $-90°(q - p)$. En cualquier sistema la pendiente de la curva del logaritmo de la magnitud en $\omega = \infty$ es igual a $-20(q - p)$ dB/década. Por tanto es posible

detectar si un sistema es de fase mínima o no examinando la pendiente de la asíntota de altas frecuencias de la curva del logaritmo de la magnitud y el ángulo de fase en $\omega = \infty$. Si la pendiente de la curva del logaritmo de la magnitud cuando ω se aproxima a infinito es $-20(q - p)$ dB/década y el ángulo de fase en $\omega = \infty$ es igual a $-90°(q - p)$, el sistema es de fase mínima.



Los sistemas de fase no mínima son lentos en su respuesta debido a su defectuoso comportamiento al comienzo de la misma. En la mayoría de los sistemas de control prácticos, se deberían de evitar cuidadosamente los excesivos retardos de fase. En el diseño de un sistema, si lo importante es la velocidad de respuesta, no se deberían utilizar componentes de fase no mínima.

Diagrama de Nyquist

El diagrama de Nyquist de una función de transferencia senoidal $G(j\omega)$ es una representación de la magnitud de $G(j\omega)$ frente al ángulo de fase de $G(j\omega)$ en coordenadas polares cuando ω varía de cero a infinito. De este modo el diagrama polar es el lugar de los vectores $|G(j\omega)|\angle G(j\omega)$ cuando se varía ω de cero a infinito. Conviene observar que en los diagramas polares un ángulo de fase positivo (negativo) se mide en dirección contraria a las agujas de un reloj (en el sentido de las agujas de un reloj) desde el eje real positivo. El diagrama de Nyquist se llama a menudo el diagrama polar. En la Figura 6-13 se muestra un ejemplo de un diagrama de este tipo. Cada punto del diagrama polar de $G(j\omega)$ representa el punto terminal de un vector en un valor particular de ω. Las proyecciones de $G(j\omega)$ sobre los ejes real e imaginario son sus componentes real e imaginaria.

Una ventaja de utilizar un diagrama de Nyquist es que muestra las características de la respuesta en frecuencia de un sistema a lo largo de todo el rango de frecuencias en una única gráfica. Una desventaja es que la representación no indica claramente la contribución de cada factor individual de la función de transferencia en lazo abierto.

En la Figura 6-14a se muestran las formas generales de las porciones de baja frecuencia de los diagramas de Nyquist de sistemas de fase mínima de tipo 0, tipo 1 y tipo 2. Se puede ver que si el grado del polinomio del denominador de $G(j\omega)$ es mayor que el del numerador, entonces el lugar de $G(j\omega)$ converge en el sentido de las agujas de un reloj al origen. En $\omega = \infty$ los lugares son tangentes a uno o a otro eje tal como se muestra en la Figura 6-14b.

Para el caso donde los grados de los polinomios del denominador y del numerador de $G(j\omega)$ son iguales, el diagrama de Nyquist comienza a una distancia finita del eje real y finaliza también en un punto finito sobre el eje real.

Observe que cualquier forma complicada en las curvas de Nyquist están originadas por la dinámica del numerador, esto es, por las constantes de tiempo en el numerador de la función de transferencia.

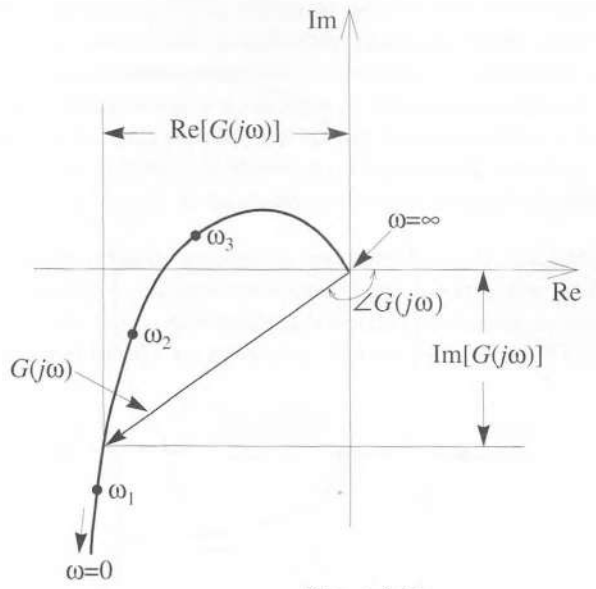


Figura 6-13

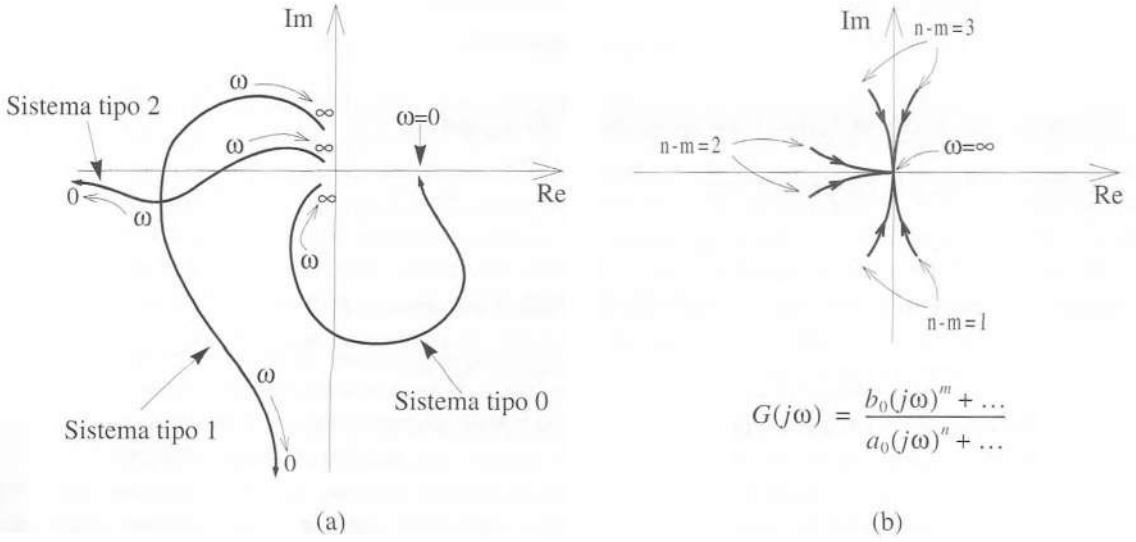


Figura 6-14

Criterio de estabilidad de Nyquist

Sea el sistema que se muestra en la Figura 6-15. La función de transferencia en lazo cerrado es

$$\frac{C(s)}{R(s)} = \frac{G(s)}{1 + G(s)H(s)}$$

Para que sea estable, todas las raíces de la ecuación característica

$$1 + G(s)H(s) = 0$$

deben permanecer en el semiplano izquierdo s. El criterio de estabilidad de Nyquist relaciona la respuesta en frecuencia en lazo abierto de $G(j\omega)H(j\omega)$ con el número de ceros y polos de $1 + G(j\omega)H(j\omega)$ que se encuentran

en el semiplano derecho s. Este criterio, que se debe a H. Nyquist, es útil en ingenieria de control porque la estabilidad absoluta del sistema en lazo cerrado se puede determinar gráficamente a partir de las curvas de respuesta en frecuencia en lazo abierto y no hay necesidad de determinar realmente los polos en lazo cerrado. Las curvas de respuesta en frecuencia en lazo abierto obtenidas analíticamente así como las que se determinan de forma experimental se pueden utilizar para el análisis de la estabilidad. Esto es conveniente porque al diseñar un sistema de control, a menudo sucede que las expresiones matemáticas de algunas de las componentes no se conocen; solamente se tienen disponibles los datos de la respuesta en frecuencia.

El criterio de estabilidad de Nyquist se puede expresar de la forma siguiente:

Criterio de estabilidad de Nyquist: En el sistema que se muestra en la Figura 6-15, si la función de transferencia en lazo abierto $G(s)H(s)$ tiene P polos en el semiplano derecho s, entonces para que sea estable el lugar $G(s)H(s)$, cuando un punto representativo s recorre el camino de Nyquist en el sentido de las agujas de un reloj, debe enlazar al punto $-1 + j0$ P veces en el sentido contrario a las agujas de un reloj.

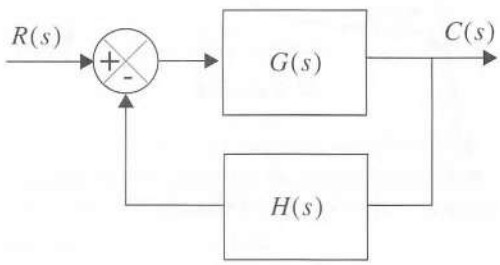


Figura 6-15

Comentarios sobre el criterio de estabilidad de Nyquist

1. Este criterio se puede expresar como

$$Z = N + P \tag{6-7}$$

donde Z = número de ceros de $1 + G(s)H(s)$ en el semiplano derecho s
N = número de vueltas en el sentido de las agujas de un reloj del punto $-1 + j0$
P = número de polos de $G(s)H(s)$ en el semiplano derecho s

Si P no es cero, para un sistema de control estable, debemos tener $Z = 0$ o $N = -P$ lo que significa que debemos tener P vueltas alrededor del punto $-1 + j0$ en el sentido contrario a las agujas de un reloj.

Si $G(s)H(s)$ no tiene ningun polo en el semiplano derecho s, entonces teniendo en cuenta (6-7), debemos tener $Z = N$ para garantizar la estabilidad. Por ejemplo, considere el sistema con la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{K}{s(T_1 s + 1)(T_2 s + 1)}$$

La Figura 6-16 muestra el camino de Nyquist y el lugar de $G(s)H(s)$ para un valor pequeño y otro grande de la ganancia K. Como el número de polos de $G(s)H(s)$ en el semiplano derecho es cero, para que este sistema sea estable, es necesario que $N = Z = 0$ o que el lugar $G(s)H(s)$ no enlace al punto $-1 + j0$.

Para valores pequeños de K, el punto $-1 + j0$ no queda enlazado. Por tanto el sistema es estable para valores pequeños de K. Para valores grandes de K, el lugar de $G(s)H(s)$ enlaza al punto $-1 + j0$ dos veces en el sentido de las agujas de un reloj, lo que indica que hay dos polos en lazo cerrado en el semiplano derecho s y que el sistema es inestable. Para obtener una buena precisión K debería ser grande. Sin embargo desde el punto de vista de estabilidad, un valor grande de K origina una pobre estabilidad o incluso inestabilidad. Para llegar a un compromiso entre precisión y estabilidad, es necesario insertar una red de compensación en el sistema.

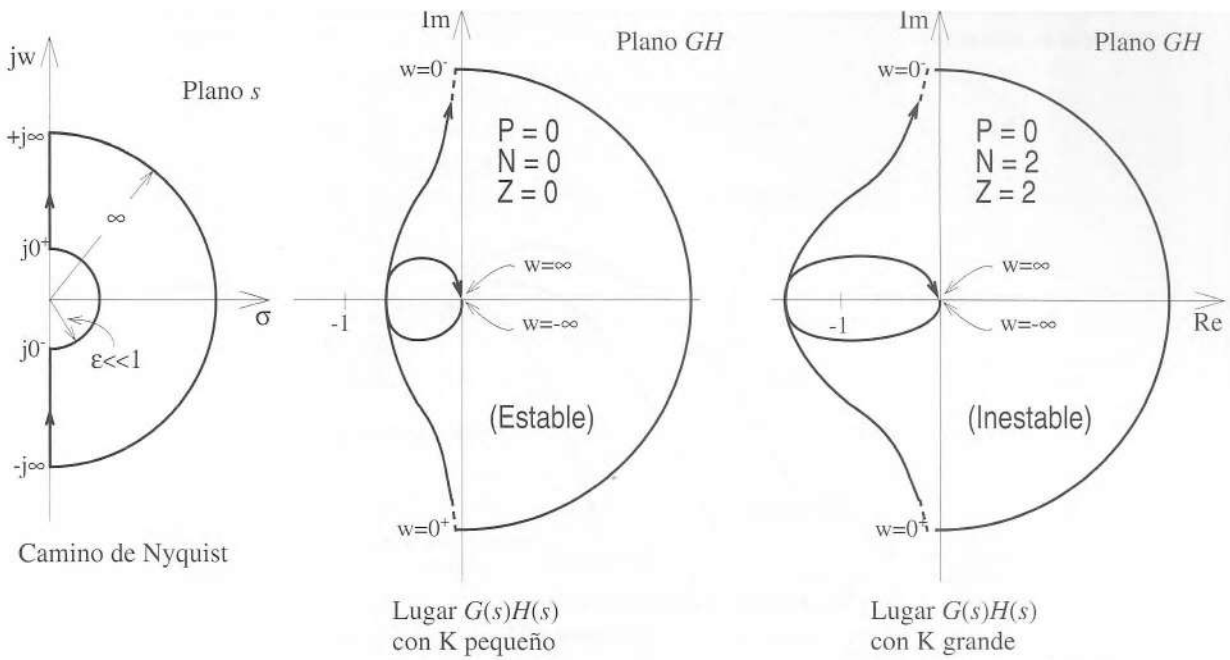


Figura 6-16

2. Debemos ser cuidadosos cuando se comprueba la estabilidad de sistemas con múltiples lazos ya que pueden incluir polos en el semiplano derecho s. (Observe que aunque un lazo interior puede ser inestable, el sistema completo en lazo cerrado puede hacerse estable mediante un diseño adecuado). La simple inspección de las vueltas al punto $-1 + j0$ del lugar $G(j\omega)H(j\omega)$ no es suficiente para detectar inestabilidad en sistemas con múltiples lazos. En tales casos, sin embargo si existen o no algunos polos de $1 + G(s)H(s)$ que está en el semiplano derecho s se puede determinar fácilmente aplicando el criterio de Routh al denominador de $G(s)H(s)$ o realmente encontrando los polos de $G(s)H(s)$ utilizando MATLAB.

3. Si el lugar de $G(j\omega)H(j\omega)$ pasa a través de del punto $-1 + j0$, entonces los ceros de la ecuación característica, o los polos en lazo cerrado están localizados sobre el eje $j\omega$. Esto no es deseable para sistemas de control prácticos. Para un sistema en lazo cerrado bien diseñado, ninguna de las raíces de la ecuación característica deberían permanecer sobre el eje $j\omega$.

Márgenes de ganancia y fase

La Figura 6-17 muestra los diagramas de Nyquist de $G(j\omega)$ para tres valores diferentes de la ganancia en lazo abierto K. Para un valor grande de la ganancia K, el sistema es inestable. Cuando la ganacia disminuye a un cierto valor, el lugar de $G(j\omega)$ pasa a través del punto $-1 + j0$. Esto significa que con este valor de la ganancia el sistema está en la frontera de inestabilidad y que el sistema exhibirá una oscilación sostenida. Para un valor pequeño de la ganancia K, el sistema es estable.

En general , cuanto más próximo se encuentre el lugar de $G(j\omega)$ para enlazar al punto $-1 + j0$ más oscilatoria será la respuesta. La cercanía del lugar $G(j\omega)$ al punto $-1 + j0$ se puede utilizar como una medida del margen de estabilidad. (Esto, sin embargo, no se cumple en el caso de los sistemas condicionalmente estable). Es una práctica común representar la proximidad en términos de los márgenes de ganancia y de fase.

Margen de fase

El margen de fase es la cantidad de retardo de fase adicional en la frecuencia de la ganancia de cruce que se requiere para llevar al sistema a su frontera de inestabilidad. La frecuencia de la ganancia de cruce es la fre-

cuencia en la cual la magnitud de la función de transferencia en lazo abierto, $|G(jw)|$, vale la unidad. El margen de fase γ es 180° mas el ángulo de fase ϕ de la función de transferencia en lazo abierto en la frecuencia de la ganancia de cruce, o

$$\gamma = 180° + \phi$$

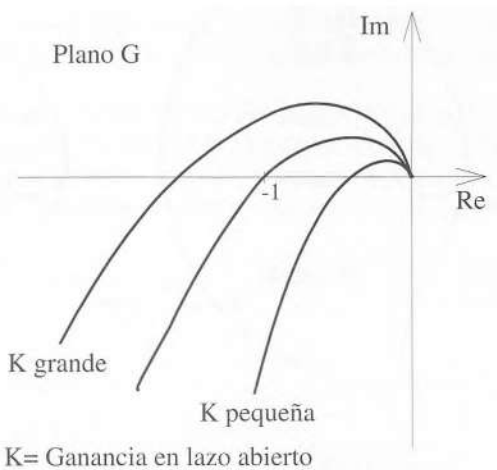


Figura 6-17

En el diagrama de Nyquist, se puede dibujar una línea desde el origen al punto en el que el círculo unidad atraviesa al lugar $G(j\omega)$. El ángulo que va desde el eje real negativo a esta línea es el margen de fase. El margen de fase es positivo para $\gamma>0$ y negativo para $\gamma<0$. Para que un sistema de fase mínima sea estable, el margen de fase debe ser positivo.

Las Figuras 6-18(a) y (b) ilustran el margen de fase de sistemas estables e inestables en los diagramas de Nyquist y de Bode. En los diagramas de Bode, el punto crítico en el plano complejo corresponde a la línea de 0 dB y a la línea de -180°.

Margen de ganancia

El margen de ganancia es el recíproco de $|G(j\omega)|$ en la frecuencia donde el ángulo de fase es -180°. Si se define la frecuencia de cruce de la fase como la frecuencia en la cual el ángulo de fase de la función de transferencia en lazo abierto es igual a -180°, el margen de ganancia K_g se puede expresar como

$$K_g = \frac{1}{|G(j\omega_1)|}$$

En términos de decibelios

$$K_g \text{ dB} = 20 \log K_g = -20 \log |G(j\omega_1)|$$

El margen de ganancia expresado en decibelios es positivo si K_g es mayor que la unidad y negativo si K_g es menor que la unidad. Así un margen de ganancia positivo (en decibelios) significa que el sistema es estable y un margen de ganancia negativo que es inestable. En las Figuras 6-18(a) y (b) se muestra el margen de ganancia.

Para un sistema estable de fase mínima, el margen de ganancia indica cuánta ganancia se puede aumentar antes de que se haga inestable el sistema. Para un sistema inestable, el margen de ganancia es indicativo de cuánta ganancia se debe disminuir para hacer estable el sistema.

Los márgenes de ganancia de los sistemas de fase mínima de primer y segundo orden son infinitos puesto que los diagramas de Nyquist de tales sistemas no cortan al eje real negativo. Así pues esta clase de sistemas no se pueden inestabilizar.

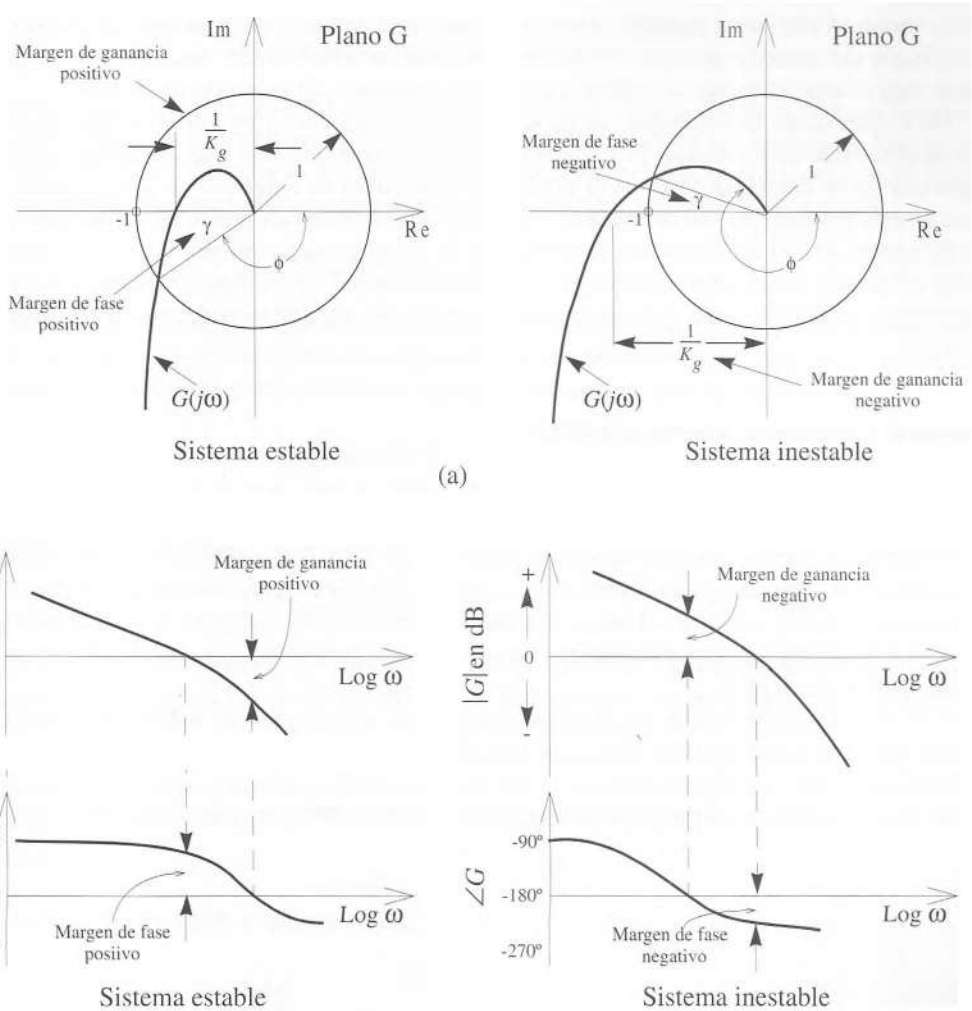


Figura 6-18

Es importante apuntar que para un sistema de fase no mínima la condición de estabilidad no se cumplirá a menos que el diagrama de $G(j\omega)$ enlace al punto $-1 + j0$. Por tanto un sistema de fase no mínima estable tendrá márgenes de fase y de ganancia negativos.

Algunos comentarios sobre los márgenes de ganancia y de fase

Los márgenes de fase y de ganancia de un sistema de control son una medida de la proximidad al punto $-1 + j0$ del diagrama de Nyquist. Por tanto estos márgenes se pueden utilizar como criterio de diseño.

Debería observarse que ni el margen de ganancia ni el margen de fase por si solos no dan una condición suficiente de estabilidad relativa. Ambos deben darse conjuntamente para la determinación de la estabilidad relativa.

Para un sistema de fase mínima ambos márgenes de ganancia y de fase deben ser positivos para que el sistema sea estable. Márgenes negativos indican inestabilidad.

Márgenes de fase y de ganancia adecuados aseguran la estabilidad del sistema frente a variaciones en las componentes del sistema. Para un comportamiento satisfactorio, el margen de fase debería estar comprendido entre 30° y 60° y el margen de ganancia ser mayor que 6 dB. Con estos valores, un sistema de fase mínima

tiene garantizada la estabilidad incluso si la ganancia en lazo abierto y las constantes de tiempo de los componentes varían en una cierta cantidad. Aunque los márgenes de fase y de ganancia dan solamente una estimación aproximada del amortiguamiento efectivo del sistema en lazo cerrado, representan una herramienta conveniente para diseñar sistemas de control o ajustar las constantes de ganancia del sistema.

Para sistemas de fase mínima, las características de magnitud y de fase de la función de transferencia en lazo abierto están relacionadas. El requisito de que el margen de fase esté comprendido entre 30° y 60° significa que en un diagrama de Bode la pendiente de la curva de magnitud en la frecuencia de la ganancia de cruce es más gradual que -40 dB/década. En la mayoría de los casos prácticos, es deseable para la estabilidad una pendiente de -20 dB/década en la frecuencia de la ganancia de cruce. Si es de -40 dB/década, el sistema podría ser estable o inestable. (Incluso si el sistema es estable, sin embargo el margen de fase es pequeño). Si la pendiente en la frecuencia de la ganancia de cruce es de -60 dB/década o más pronunciada, el sistema será inestable.

Sistemas condicionalmente estables

La Figura 6-19 muestra un ejemplo de un lugar $G(j\omega)H(j\omega)$ para el cual el sistema en lazo cerrado puede hacerse estable o inestable variando la ganancia en lazo abierto. Si la ganancia en lazo abierto se aumenta suficientemente, el lugar $G(j\omega)H(j\omega)$ encierra al punto $-1 + j0$ dos veces y el sistema se hace inestable. Si la ganancia en lazo abierto disminuye suficientemente, otra vez el lugar $G(j\omega)H(j\omega)$ encierra al punto $-1 + j0$ dos veces. El sistema es estable sólo para el rango limitado de valores de la ganancia en lazo abierto para los cuales el punto $-1 + j0$ permanece completamente fuera del lugar $G(j\omega)H(j\omega)$. Un sistema de este tipo se dice que es condicionalmente estable.

Esta clase de sistemas se hacen inestables cuando se aplican grandes señales de entradas ya que se puede originar una saturación que tiene como consecuencia una reducción de la ganancia en lazo abierto del sistema.

Para garantizar una operación estable del sistema condicionalmente estable considerado aquí, el punto crítico -1 + j0 no debe localizarse en las regiones entre OA y BC que se muestran en la Figura 6-19.

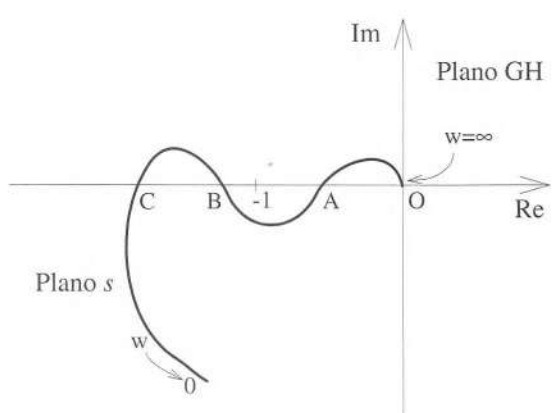


Figura 6-19

6.3 REPRESENTACIÓN DEL DIAGRAMA DE BODE CON MATLAB

La orden bode calcula las magnitudes y los angulos de fase de la respuesta en frecuencia de sistemas continuos, lineales e invariantes en el tiempo. Como se expuso con anterioridad, los diagramas de Bode se utilizan frecuentemente para analizar y diseñar sistemas de control. Los diagramas de Bode indican el margen de ganancia, el margen de fase, la ganancia DC, el ancho de banda, etc.

Cuando se introduce la orden bode (sin argumentos en el lado izquierdo) en el computador, MATLAB realiza el diagrama de Bode sobre la pantalla.

Cuando se llama con argumentos en el lado izquierdo,

[mag,fase,w] = bode(num,den,w)

bode devuelve la respuesta en frecuencia del sistema en las matrices mag, fase y w. Ningún diagrama es dibujado sobre la pantalla. Las matrices mag y fase contienen las magnitudes y los ángulos de fase de la respuesta en frecuencia del sistema evaluados en los puntos de frecuencia especificados por el usuario. El ángulo de fase se devuelve en grados. La magnitud se puede convertir a decibelios mediante la declaración

magdB = 20*log10(mag)

Para especificar el rango de frecuencias, utilice la orden logspace(d1,d2) o logspace(d1,d2,n). logspace(d1,d2) genera un vector de 50 puntos espaciados logarítmicamente por igual entre las décadas 10^{d1} y 10^{d2}. Es decir, para generar 50 puntos entre 0.1 rad/seg y 100 rad/seg, introduzca la orden

w = logspace(-1,2)

logspace(d1,d2,n) genera n puntos espaciados logarítmicamente por igual entre las décadas 10^{d1} y 10^{d2}. Por ejemplo, para generar 100 puntos entre 1 rad/seg y 1000 rad/seg, introduzca la siguiente orden:

w = logspace(0,3,100)

Para incorporar estos puntos de frecuencia al dibujar los diagramas de Bode, utilice la orden bode(num,den,w) o bode(A,B,C,D,iu,w). Estas órdenes utilizan el vector de frecuencias w especificado por el usuario.

EJEMPLO 6-1

Considere la siguiente función de transferencia:

$$\frac{Y(s)}{U(s)} = \frac{25}{s^2 + 4s + 25}$$

Dibuje un diagrama de Bode para esta función de transferencia.

Cuando el sistema viene definido de la forma

$$G(s) = \frac{\text{num}(s)}{\text{den}(s)}$$

utilice la orden bode(num,den) para dibujar el diagrama de Bode. [Cuando el numerador y el denominador contienen los coeficientes de polinomio en potencias descendientes de s, bode(num,den) dibuja el diagrama de Bode]. El programa de MATLAB 6-1 muestra un programa para dibujar el diagrama de Bode para este sistema. El diagrama de Bode resultante se muestra en la Figura 6-20.

Programa en MATLAB 6-1

```
num=[0 0 25];
den=[1 4 25];
bode(num,den)
title('Diagrama de Bode de G(s)=25/(s^2+4s+25)')
```

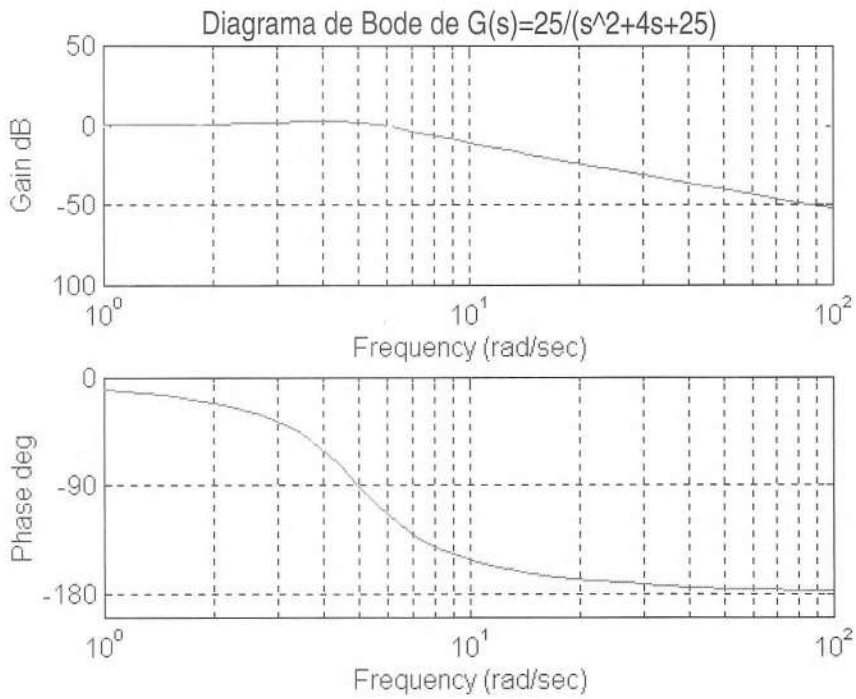


Figura 6-20

EJEMPLO 6-2

Considere el sistema mostrado en la Figura 6-21. La función de transferencia en lazo abierto es

$$G(s) = \frac{9(s^2 + 0{,}2s + 1)}{s(s^2 + 1{,}2s + 9)}$$

Dibuje un diagrama de Bode.

El programa de MATLAB 6-2 dibuja el diagrma de Bode para el sistema. El dibujo resultante se muestra en la Figura 6-22. El rango de frecuencias en este caso se determina automáticamente y va desde 0.1 rad/seg a 10 rad/seg.

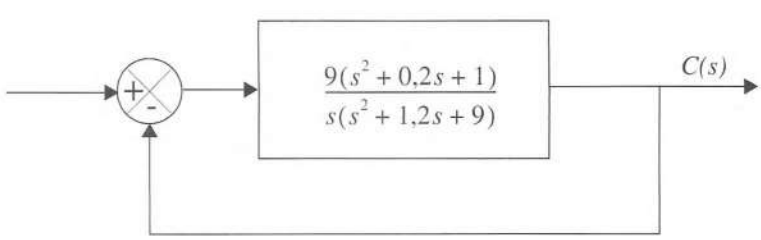


Figura 6-21

Programa en MATLAB 6-2

```
num=[0 9 1.8 9];
den=[1 1.2 9 0];
bode(num,den)
title('Diagrama de Bode de G(s)=9(s^2+0.2s+1)/[s(s^2+1.2s+9)]')
```

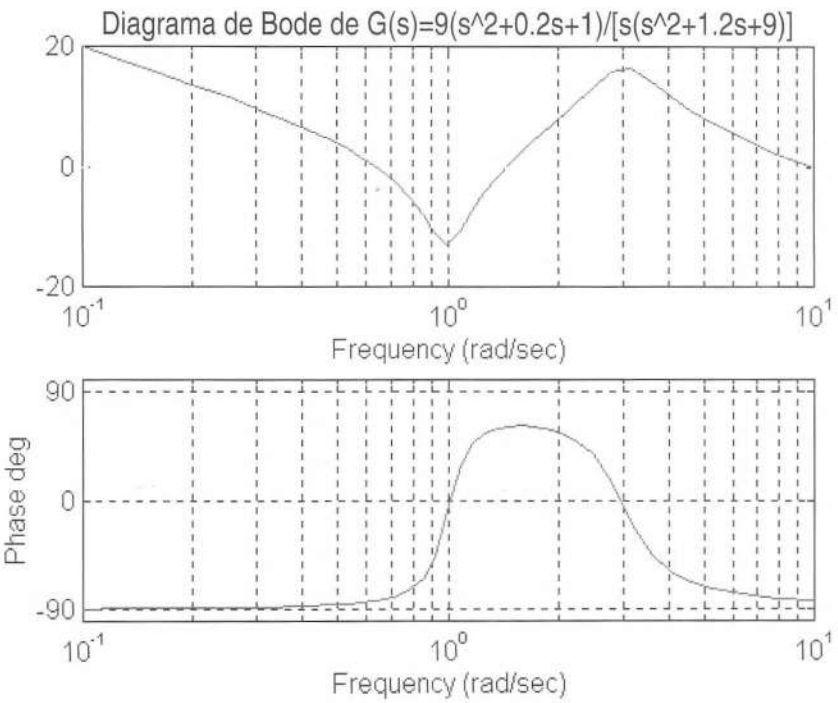


Figura 6-22

Si se desea dibujar el diagrama de Bode desde 0.01 rad/seg hasta 1000 rad/seg, introduzca la siguiente orden:

w = logspace(-2,3,100)

Esta orden genera 100 puntos espaciados logarítmicamente por igual entre 0.01 rad/seg y 1000 rad/seg. (Observe que el vector w especifica las frecuencias en radianes por segundo en las cuales se calculará la respuesta en frecuencia).

Si se utiliza la orden

bode(num,den,w)

el rango de frecuencia será especificado por el usuario, mientras que el rango de la magnitud y el del ángulo de fase se determinará automáticamente. Véase el programa de MATLAB 6-3 y el gráfico resultante en la Figura 6-23.

Para especificar los rangos de magnitud y ángulo de fase, utilice la siguiente orden:

[mag,fase,w] = bode(num,den,w)

Programa en MATLAB 6-3

```
num=[0 9 1.8 9];
den=[1 1.2 9 0];
w=logspace(-2,3,100);
bode(num,den,w)
title('Diagrama de Bode de G(s)=9(s^2+0.2s+1)/[s(s^2+1.2s+9)]')
```

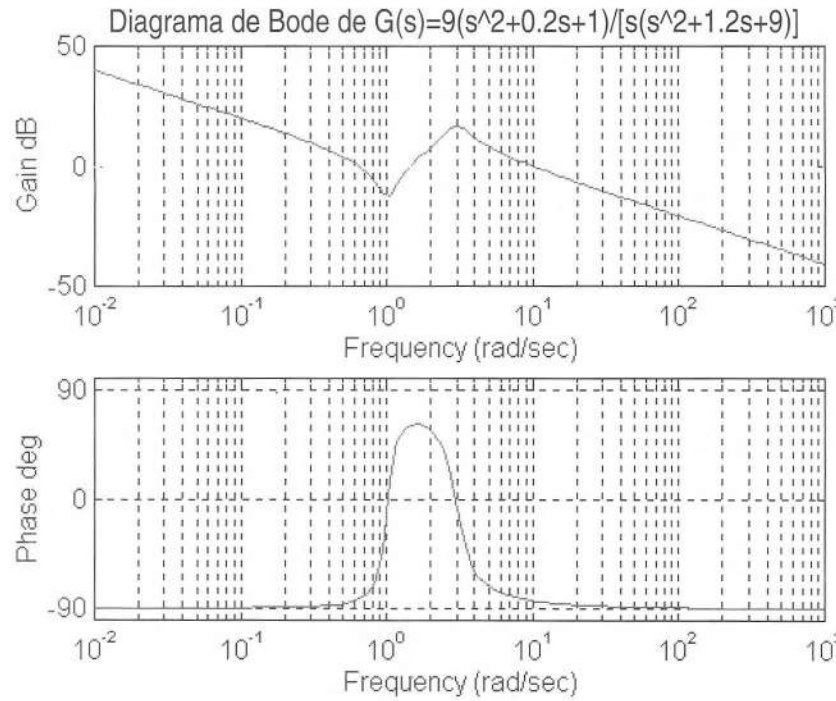


Figura 6-23

Las matrices mag y fase contienen las magnitudes y los ángulos de fase de la respuesta en frecuencia evaluada en los puntos de frecuencia especificados por el usuario. El ángulo de fase se devuelve en grados. La magnitud se puede convertir a decibelios mediante la declaración

```
magdB = 20*log10(mag)
```

Si se desea especificar el rango de magnitud para que se encuentre, por ejemplo, al menos entre -45 dB y +45 dB, introduzca líneas no visibles en -45 dB y +45 dB en el dibujo especificando dBmax (magnitud máxima) y dBmin (magnitud mínima) como sigue:

```
dBmax = 45*ones(1,100);
dBmin = -45*ones(1,100);
```

después introduzca la siguiente orden de dibujo semilog:

```
semilogx(w,magdB,'o',w,magdB,'-',w,dBmax,'--i',w,dBmin,':i')
```

(Observe que el número de puntos dBmax y el número de puntos dBmin deben de coincidir con el número de puntos de frecuencia de w. En este ejemplo, todos los puntos son 100). En la pantalla aparecerá la curva de magnitud magdB con las marcas 'o'. (Las líneas rectas en +45 dB y en -45 dB no son visibles).

Observe que 'i' es un color no visible. Por ejemplo, 'og' mostrará pequeños círculos en color verde y 'oi' mostrará pequeños círculos en color 'no visible': es decir, no se verán los pequeños círculos sobre la pantalla. Al cambiar una parte de la orden semilogx escrita arriba

```
w,dBmax,'--i',w,dBmin,':i'
```

por

```
w,dBmax,'--',w,dBmin,':'
```

la línea de +45 dB y la línea de -45 dB se harán visibles sobre la pantalla.

El rango para la magnitud es normalmente un múltiplo de 5 dB, 10 dB, 20 dB o 50 dB. (Existen excepciones). Para el caso que se considera, el rango de magnitud irá de -50 dB a +50 dB.

Para el ángulo de fase, si se desea especificar el rango para que se encuentre, por ejemplo, al menos entre -145° y +115°, introduzca líneas no visibles en -145° y +115° en el programa especificando fmax (fase máxima) y fmin (fase mínima) como sigue:

```
fmax = 115*ones(1,100)
fmin = -145*ones(1,100)
```

Despues introduzca la orden de dibujo semilog:

```
semilogx(w,fase,'o',w,fase,'-',w,fmax,'--i',w,fmin,':i')
```

(El número de puntos de fmax y el número de puntos de fmin deben de coincidir con el número de puntos de frecuencia de w). En la pantalla aparecerá la curva de fase. Las líneas rectas en +115° y en -145° no son visibles.

El rango para el ángulo de fase es normalmente un múltiplo de 5°, 10°, 50° o 100°. (Existen excepciones). Para el caso que se considera, el rango del ángulo de fase estará entre -150° y +150°.

El programa de MATLAB 6-4 produce el diagrama de Bode para el sistema tal que el rango de frecuencia va desde 0.01 rad/seg hasta 1000 rad/seg, el rango de magnitud va desde -50 dB hasta +50 dB (el rango de magnitud es un múltiplo de 50 dB) y el rango del ángulo de fase de -150° hasta +150° (el rango del ángulo de fase es un múltiplo de 50°). La Figura 6-24 muestra el diagrama de Bode que se obtiene al ejecutar el programa de MATLAB 6-4.

Programa en MATLAB 6-4

```
%-------Diagrama de Bode-------

%*****En este programa se obtendrá el diagrama de Bode de un sistema descrito
%por su función de transferencia y se utilizará un rango de frecuencias
%especificado por el usuario*****

%*****Introduzca el numerador y el denominador de la función de
%transferencia*****

num=[0 9 1.8 9];
den=[1 1.2 9 0];
```

```
%*****Especifique el rango de frecuencias e introduzca la orden
%[mag,fase,w]=bode(num,den,w)*****

w=logspace(-2,3,100);
[mag,fase,w]=bode(num,den,w);

%*****Se convierte la magnitud a decibelios*****

magdB=20*log10(mag);

%*****Especifique el rango para la magnitud. Para el sistema considerado, el
%rango de la magnitud debería incluir -45 dB y +45 dB. Introduzca dBmax y dBmin
%en el programa y dibuje las líneas dBmax y dBmin en un color no visible. Para
%representar la curva magdB y las líneas no visibles, introduzca lo siguiente:
%dBmax, dBmin y la orden semilogx*****

dBmax=45*ones(1,100);
dBmin=-45*ones(1,100);
semilogx(w,magdB,'o',w,magdB,'-',w,dBmax,'--i',w,dBmin,':i')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
%x e y*****

grid
title('Diagrama de Bode de G(s)=9(s^2+0.2s+1)/[s(s^2+1.2s+9)]')
xlabel('Frecuencia (rad/seg)')
ylabel('Ganancia dB')

%*****A continuación, se representa la curva de fase*****

%*****Especifique el rango para la fase. Para el sistema considerado, el rango
%de fase debería incluir -145 grados y +115 dB. Introduzca fmax y fmin en el
%programa y dibuje las líneas fmax y fmin en un color no visible. Para
%representar la curva de fase y las líneas no visibles, introduzca lo siguiente
%fmax, fmin y la orden semilogx*****

fmax=115*ones(1,100);
fmin=-145*ones(1,100);
semilogx(w,fase,'o',w,fase,'-',w,fmax,'--i',w,fmin,':i')

%*****Introduzca la rejilla, el título de la gráfica y las etiquetas de los ejes
x e y*****

grid
xlabel('Frecuencia (rad/seg)')
ylabel('Fase grados')
```

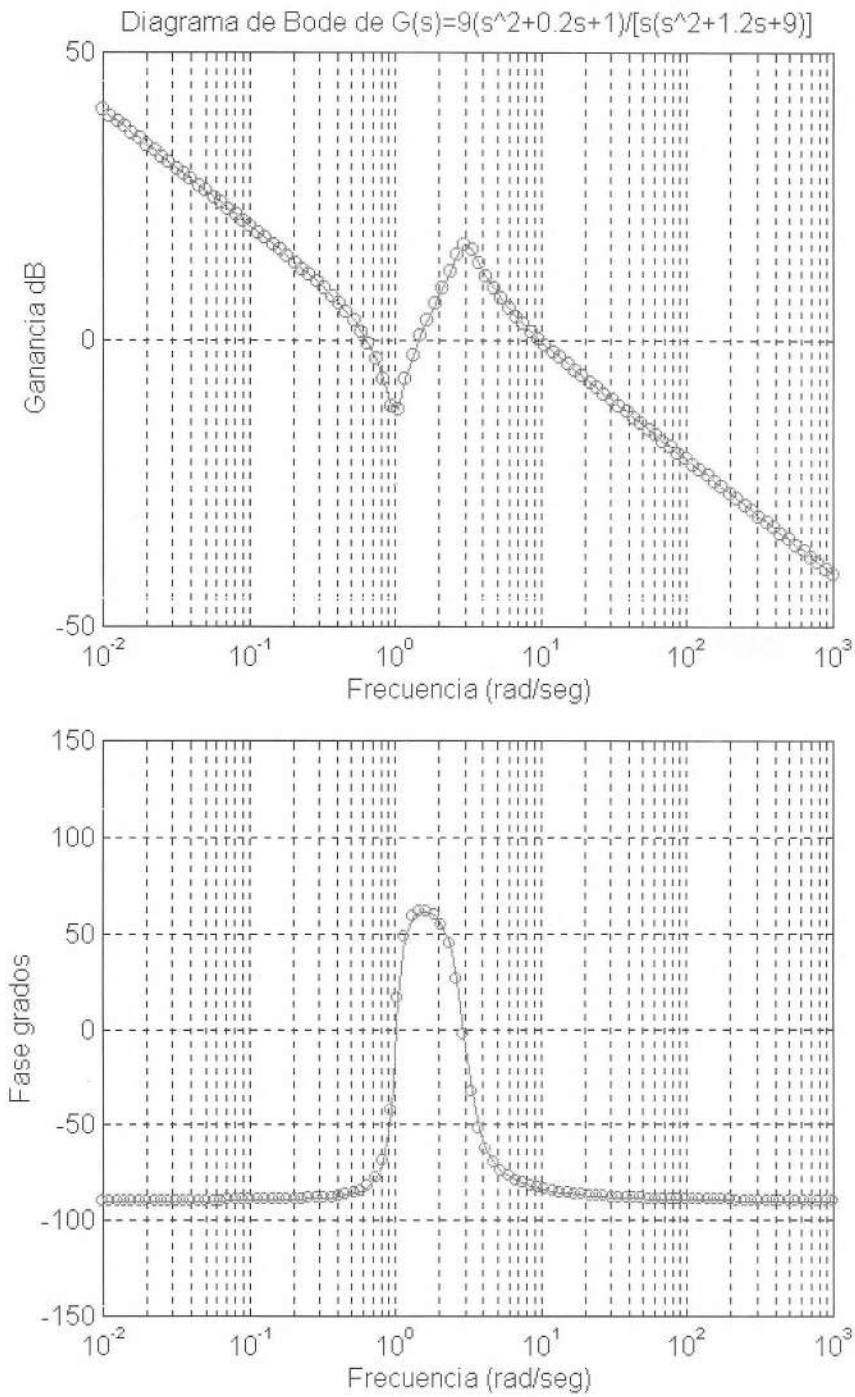


Figura 6-24

Qué sucede en el diagrama de Bode si la ganancia se hace infinita a una cierta frecuencia

Si hay un polo del sistema sobre el eje $j\omega$ y ocurre que el vector w contiene este punto de frecuencia, la ganancia se hace infinita a esta frecuencia. En tal caso, MATLAB produce mensajes de aviso (warnings). Considere el siguiente ejemplo.

EJEMPLO 6-3

Considere un sistema con la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{1}{s^2 + 1}$$

Esta función de transferencia en lazo abierto tiene polos sobre el eje $j\omega$ en $\pm j$.

Se puede utilizar el programa de MATLAB 6-5 para dibujar el diagrama de Bode de este sistema. El gráfico resultante se muestra en la Figura 6-25. Teóricamente, la magnitud llega a ser infinita en el punto de frecuencia donde w = 1 rad/seg. Sin embargo, este punto de frecuencia no se encuentra entre los puntos de frecuencia calculados. En el gráfico se muestra que el pico de magnitud alcanza aproximadamente los 50 dB. este valor está calculado cerca de, pero no exactamente en, w = 1 rad/seg.

Sin embargo, si uno de los puntos de frecuencias calculados coincidiera con el polo en w = 1, la magnitud llegaría a ser infinita en este punto. MATALAB enviaría mensajes de aviso. Véase el programa de MATLAB 6-6, donde los puntos calculados incluyen al punto w = 1 rad/seg. (En este caso hay 101 puntos calculados. El rango de los puntos calculados va desde w = 0.1 hasta w = 10. El punto correspondiente al lugar 50 está en w = 1).

Programa en MATLAB 6-5

```
num=[0 0 1];
den=[1 0 1];
bode(num,den)
title('Diagrama de bode de G(s)=1/(s^2+1)')
```

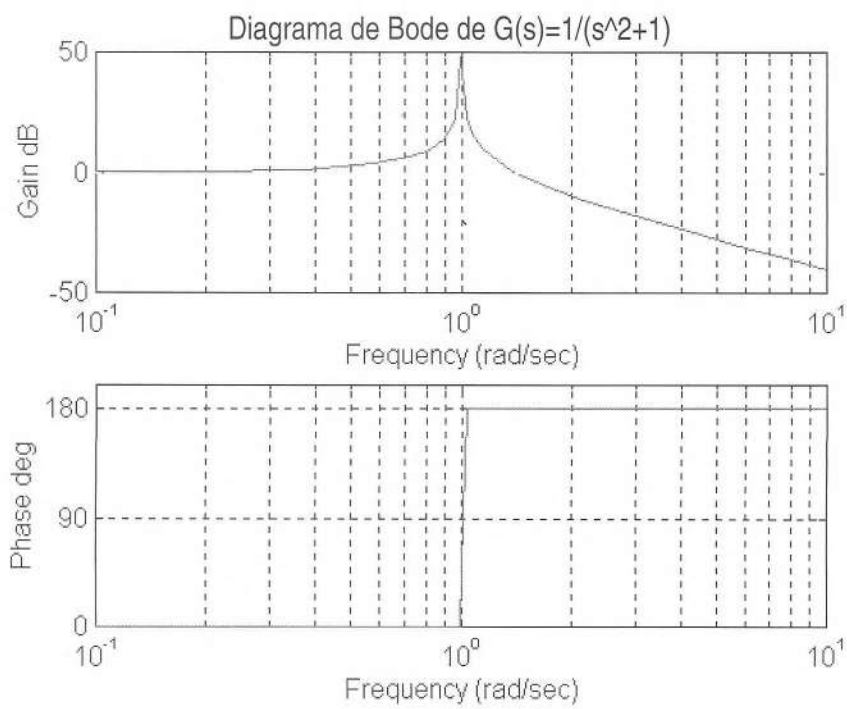


Figura 6-25

Cuando se introduce en el computador el programa de MATLAB 6-6, aparecen mensajes de aviso. El diagrama de Bode resultante, que se muestra en la Figura 6-26, no incluye la magnitud calculada en w = 1. (Teóricamente, esta magnitud es infinita). La curva de magnitud muestra el valor pico alrededor de 20 dB. La curva de fase muestra un cambio gradual en el ángulo de fase de 0° a +180° cerca del punto w = 1. (Teóricamente, el cambio en el ángulo de fase de 0° a +180° debería ser brusco en w = 1). Obviamente, el diagrama de Bode que se muestra en la Figura 6-26 es incorrecto.

Si el vector w contiene tal punto de frecuencia, donde la ganancia llega a ser infinito, cambie el número de puntos de frecuencia , por ejemplo, de 101 a 100. Normalmente, un pequeño cambio en el número de puntos de frecuencia evitará esta clase de problemas.

Programa en MATLAB 6-6

```
num=[0 0 1];
den=[1 0 1];
w=logspace(-1,1,101);
bode(num,den,w)
title('Diagrama de Bode incorrecto')
```

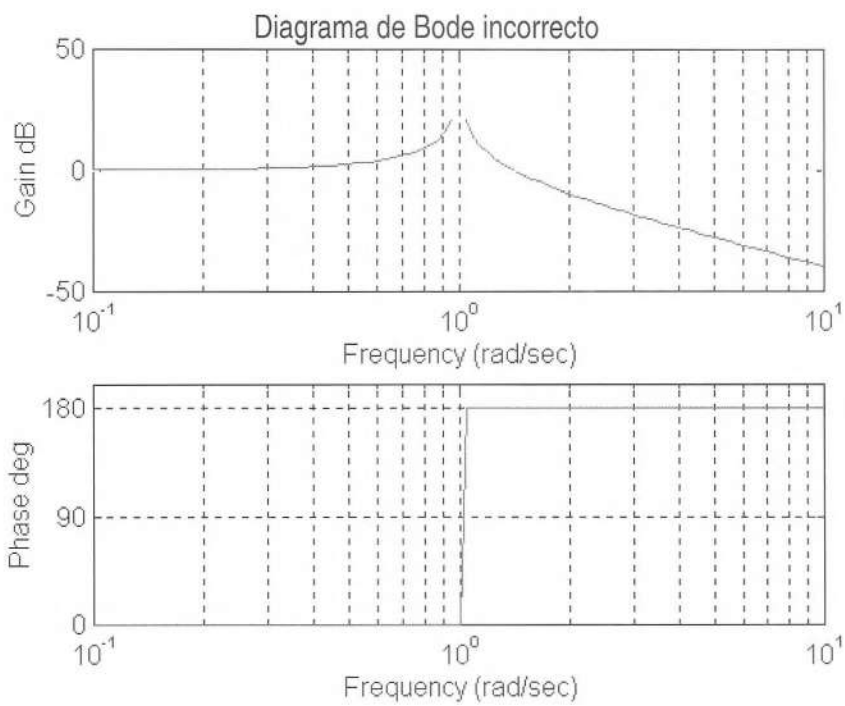


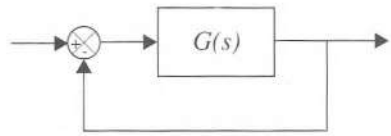
Figura 6-26

Ejemplos de diagramas de Bode

En lo que sigue, se verán algunos diagramas típicos de Bode, se incluyen sistemas de fase no mínima y sistemas con realimentación positiva. Estos diagramas se presentan en los ejemplos que van desde el 6-4 hasta el 6-10.

EJEMPLO 6-4

Sistema

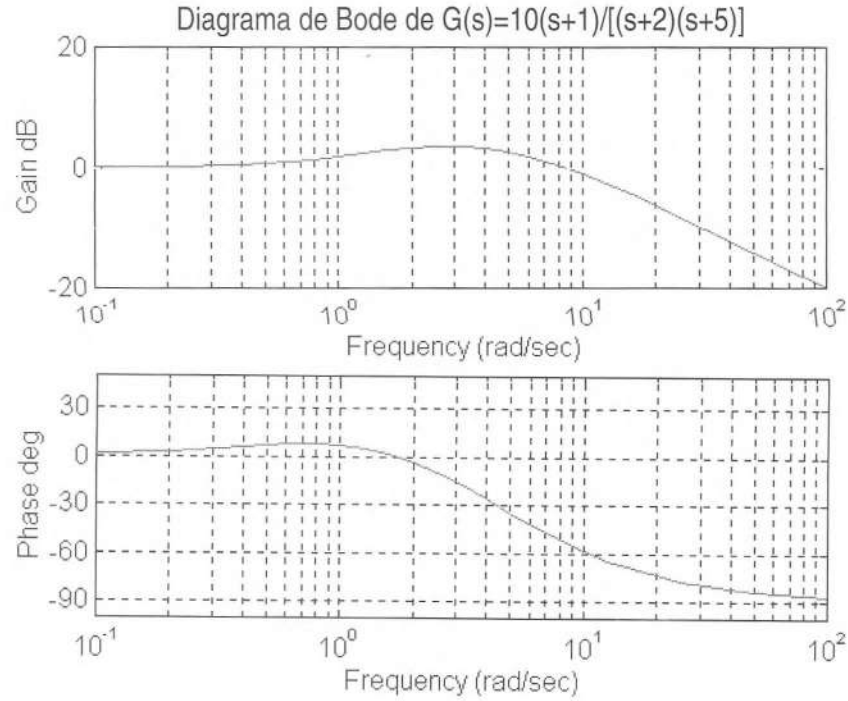


$$G(s) = \frac{10(s+1)}{(s+2)(s+5)}$$

Programa en MATLAB

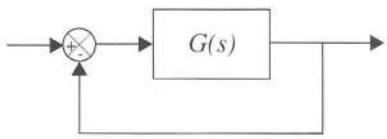
```
num=[0 10 10];
den=[1 7 10];
bode(num,den)
title('Diagrama de Bode de G(s)=10(s+1)/[(s+2)(s+5)]')
```

Diagrama de Bode



EJEMPLO 6-5

Sistema

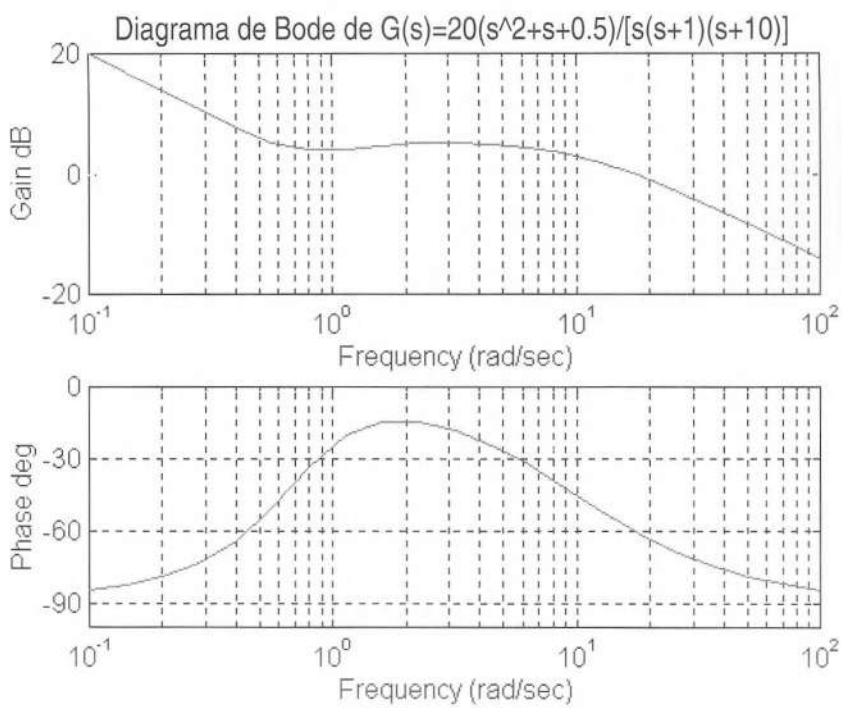


$$G(s) = \frac{20(s^2 + s + 0.5)}{s(s+1)(s+10)}$$

Programa en MATLAB

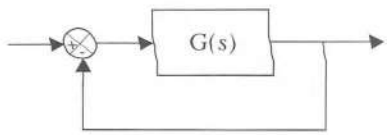
```
num=[0 20 20 10];
den=[1 11 10 0];
bode(num,den)
title('Diagrama de Bode de G(s)=20(s^2+s+0.5)/[s(s+1)(s+10)]')
```

Diagrama de Bode



EJEMPLO 6-6

Sistema

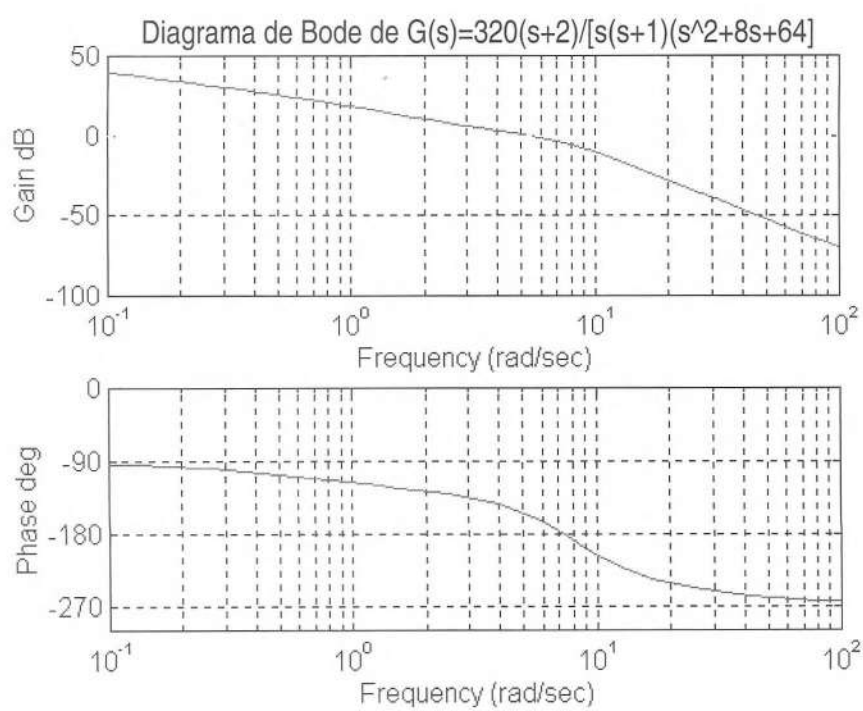


$$G(s) = \frac{K(s+2)}{s^2}$$

Programa en MATLAB

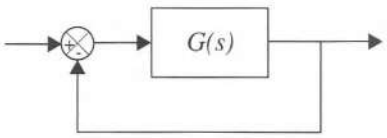
```
num=[0 0 0 320 640];
den=[1 11 72 64 0];
bode(num,den)
title('Diagrama de Bode de G(s)=320(s+2)/[s(s+1)(s^2+8s+64]')
```

Diagrama de Bode



EJEMPLO 6-7

Sistema

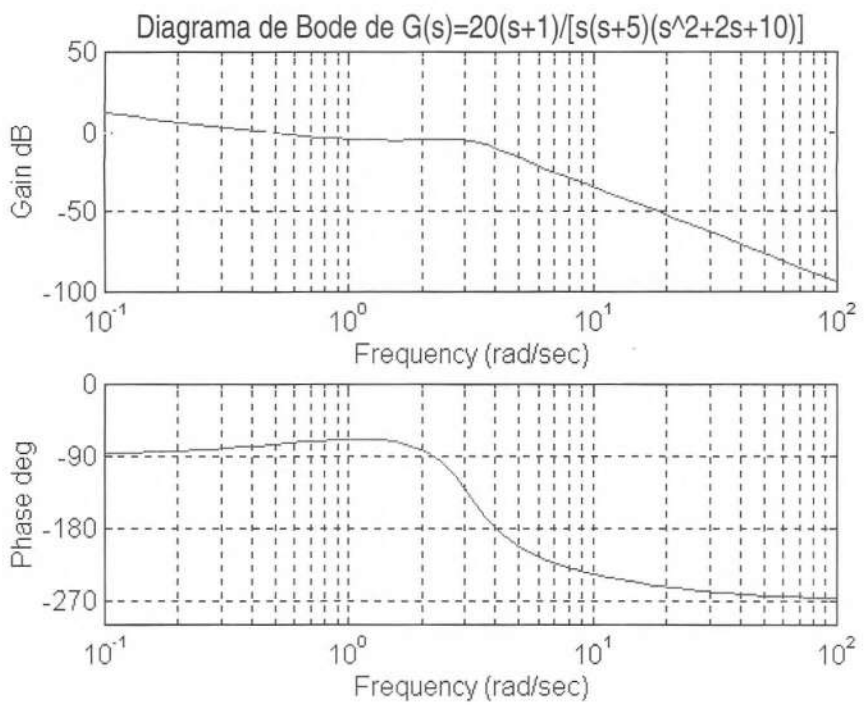


$$G(s) = \frac{20(s+1)}{s(s+5)(s^2+2s+10)}$$

Programa en MATLAB

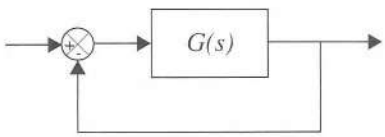
```
num=[0 0 0 20 20];
den=[1 7 20 50 0];
bode(num,den)
title('Diagrama de Bode de G(s)=20(s+1)/[s(s+5)(s^2+2s+10)]')
```

Diagrama de Bode



EJEMPLO 6-8

Sistema

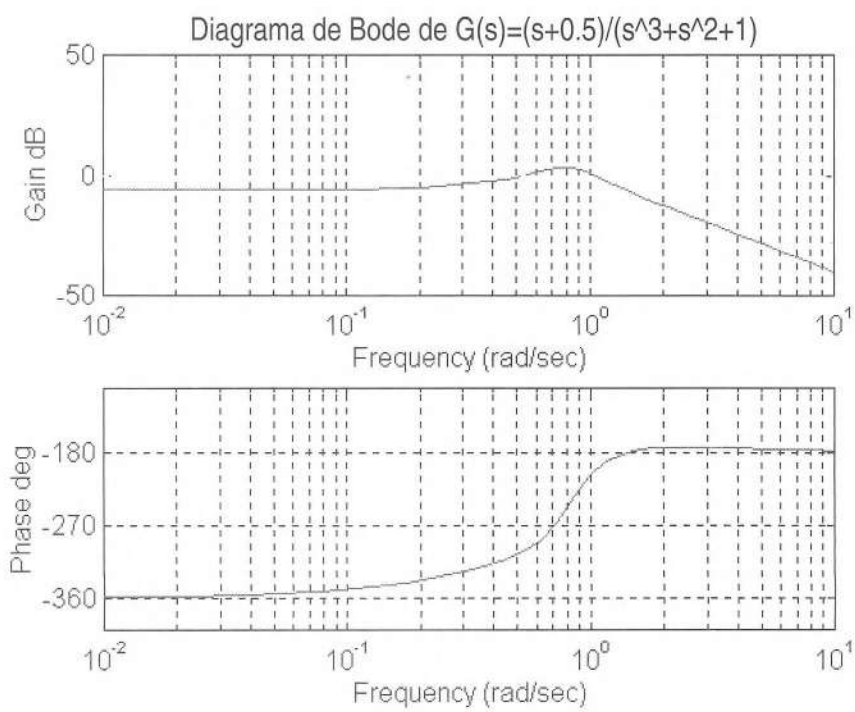


$$G(s) = \frac{s + 0.5}{s^3 + s^2 + 1}$$

Programa en MATLAB

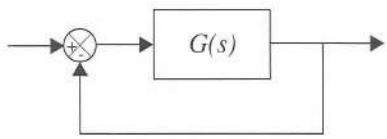
```
num=[0 0 1 0.5];
den=[1 1 0 1];
bode(num,den)
title('Diagrama de Bode de G(s)=(s+0.5)/(s^3+s^2+1)')
```

Diagrama de Bode



EJEMPLO 6-9

Sistema

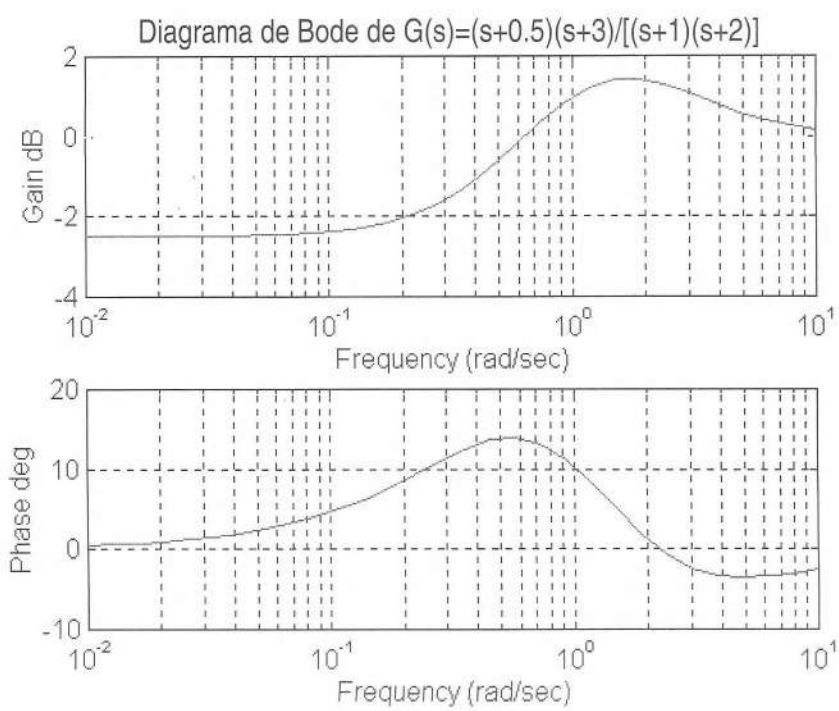


$$G(s) = \frac{s^2 + 3.5s + 1.5}{s^2 + 3s + 2}$$

Programa en MATLAB

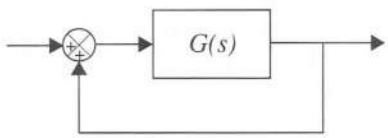
```
num=[1 3.5 1.5];
den=[1 3 2];
bode(num,den)
title('Diagrama de Bode de G(s)=(s+0.5)(s+3)/[(s+1)(s+2)]')
```

Diagrama de Bode



EJEMPLO 6-10

Sistema



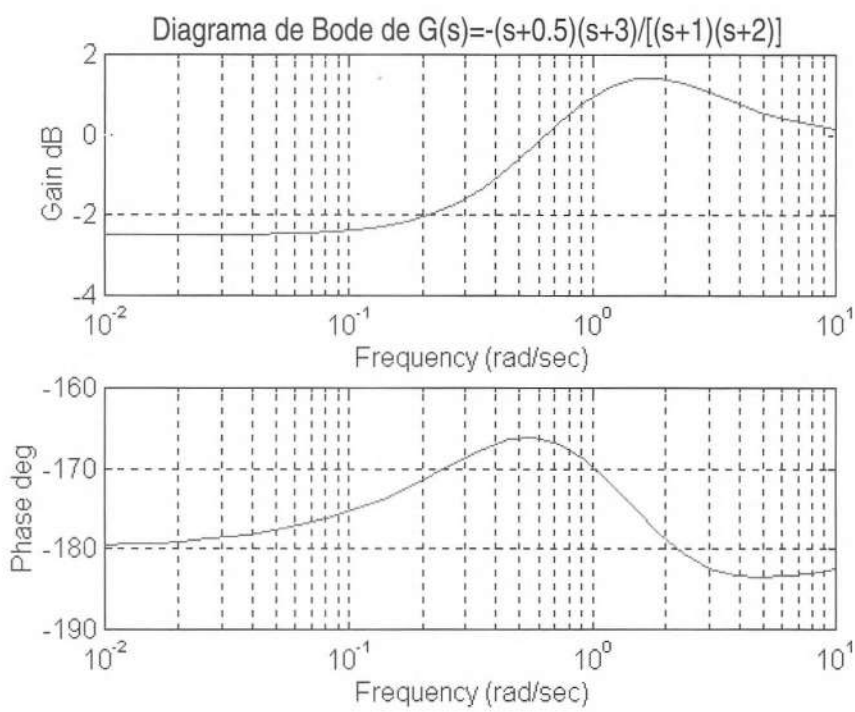
$$G(s) = \frac{s^2 + 3.5s + 1.5}{s^2 + 3s + 2}$$

Programa en MATLAB

Este es un sistema con realimentación positiva. Para convertirlo a uno con realimentación negativa, multiplicamos el numerador por -1 de forma que num = [-1.5 -3.5 -1.5]. El programa en MATLAB es

```
num=[-1 -3.5 -1.5];
den=[1 3 2];
bode(num,den)
title('Diagrama de Bode de G(s)=-(s+0.5)(s+3)/[(s+1)(s+2)]')
```

Diagrama de Bode



Obtención de diagramas de Bode de sistemas definidos en el espacio de estados

Considere el sistema definido por

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{6-8}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \tag{6-9}$$

donde

$\mathbf{x}$ = vector de estado (n-vector)
$\mathbf{y}$ = vector salida (m-vector)
$\mathbf{u}$ = vector de control (r-vector)
$\mathbf{A}$ = matriz de estado (matriz de $n \times n$)
$\mathbf{B}$ = matriz de control (matriz de $n \times r$)
$\mathbf{C}$ = matriz de salida (matriz de $m \times n$)
$\mathbf{D}$ = matriz de transmisión directa ($m \times r$)

Un diagrama de Bode para este sistema se puede obtener introduciendo la orden

```
bode(A,B,C,D)
```

o

```
bode(A,B,C,D,iu)
```

La orden bode(A,B,C,D) produce una serie de diagramas de Bode, uno por cada entrada del sistema, con el rango de frecuencias automáticamente determinado. (Un mayor número de puntos se utiliza cuando la respuesta cambia rápidamente).

La orden bode(A,B,C,D,iu), donde iu es la iésima entrada del sistema, produce los diagramas de Bode para la entrada iu a todas las salidas (y_1, y_2, ..., y_m) del sistema, con el rango de frecuencias automáticamente determinado. (El escalar iu es un índice dentro de las entradas del sistema y especifica qué entrada se va a utilizar para dibujar los diagramas de Bode). Si el vector de control $\mathbf{u}$ tiene tres entradas como

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

entonces iu debe de ser fijado en 1, 2 o 3.

Si el sistema tiene una única entrada u, se puede utilizar cualquiera de las siguientes órdenes

```
bode(A,B,C,D)
```

o

```
bode(A,B,C,D,1)
```

EJEMPLO 6-11

Considere el siguiente sistema:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Este sistema tiene una entrada u y una salida y. Utilizando la orden

bode(A,B,C,D)

e introduciendo el programa de MATLAB 6-7 en el computador, se obtiene el diagrama de Bode que se muestra en la Figura 6-27.

Si se sustituye la orden bode(A,B,C,D) en el programa de MATLAB 6-7 por

bode(A,B,C,D,1)

MATLAB producirá un diagrama de Bode idéntico al de la Figura 6-27.

Observe que si se utiliza, por error, la orden

bode(A,B,C,D,2)

Programa en MATLAB 6-7

```
A=[0 1;-25 -4];
B=[0;25];
C=[1 0];
D=[0];
bode(A,B,C,D)
title('Diagrama de Bode')
```

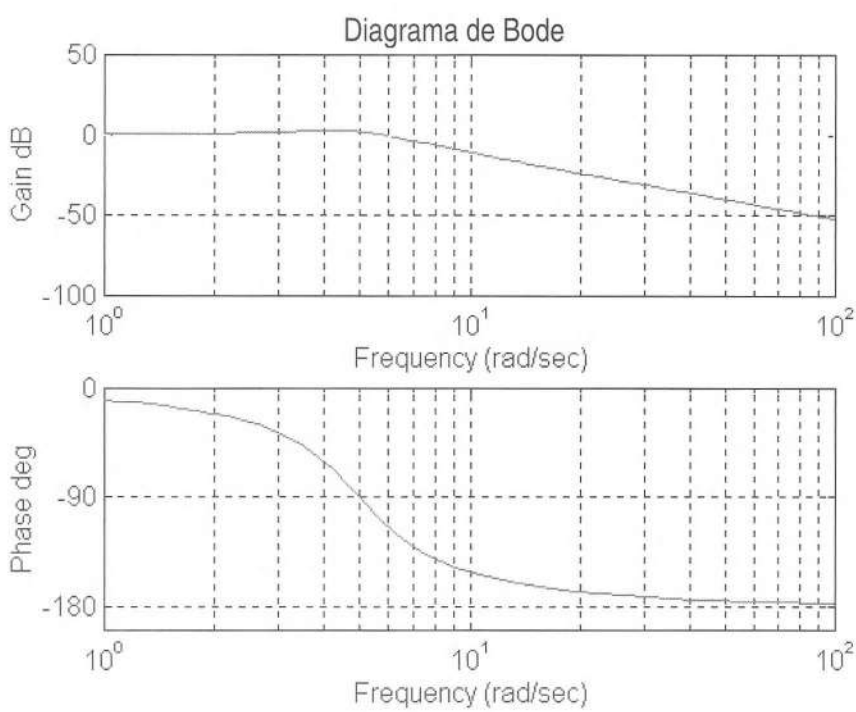


Figura 6-27

MATLAB produce un mensaje de error, ya que el sistema que se considera tiene una única entrada e iu debería fijarse a '1' y no a '2' ni a ningún otro número. Véase la siguiente salida de MATLAB:

```
A=[0 1;-25 -4];
B=[0;25];
C=[1 0];
D=[0];
bode(A,B,C,D,2)
???  Index exceeds matrix dimensions.

Error in ==> MM\MATLAB\SIGSYS\freqresp.m
On line 38  ==>     b = p' * b(:,iu);

Error in ==> MM\MATLAB\SIGSYS\bode.m
On line 74  ==> g = freqresp(a,b,c,d,iu,sqrt(-1)*w);
```

Para evitar este error, para una entrada, los sistemas de una salida deberían de utilizar la orden bode(A,B,C,D).

EJEMPLO 6-12

Considere el sistema definido por

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Obtenga las funciones de transferencia sinusoidales $Y_1(j\omega)/U_1(j\omega)$, $Y_2(j\omega)/U_1(j\omega)$, $Y_1(j\omega)/U_2(j\omega)$ y $Y_2(j\omega)/U_2(j\omega)$. Al obtener $Y_1(j\omega)/U_1(j\omega)$ y $Y_2(j\omega)/U_1(j\omega)$ suponemos que $U_2(j\omega) = 0$. Análogamente, para obtener $Y_1(j\omega)/U_2(j\omega)$ y $Y_2(j\omega)/U_2(j\omega)$ se supone que $U_1(j\omega) = 0$. Obtener también los diagramas de Bode de estas cuatro funciones de transferencia con MATLAB.

La expresión de la matriz de transferencia para el sistema definido por

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{6-10}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \tag{6-11}$$

viene dada por

$$\mathbf{Y}(s) = \mathbf{G}(s)\mathbf{U}(s) \tag{6-12}$$

donde $\mathbf{G}(s)$ es la matriz de transferencia que viene dada por

$$\mathbf{G}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$$

Para el sistema considerado aquí, la matriz de transferencia es

$$\mathbf{C}(s\mathbf{I}-\mathbf{A})^{-1}\mathbf{B}+\mathbf{D} = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}s & -1\\ 25 & s+4\end{bmatrix}^{-1}\begin{bmatrix}1 & 1\\ 0 & 1\end{bmatrix}$$

$$= \frac{1}{s^2+4s+25}\begin{bmatrix}s+4 & 1\\ -25 & s\end{bmatrix}\begin{bmatrix}1 & 1\\ 0 & 1\end{bmatrix}$$

$$= \begin{bmatrix}\dfrac{s+4}{s^2+4s+25} & \dfrac{s+5}{s^2+4s+25}\\ \dfrac{-25}{s^2+4s+25} & \dfrac{s-25}{s^2+4s+25}\end{bmatrix} \tag{6-13}$$

Así, sustituyendo (6-13) en (6-12), se obtiene

$$\begin{bmatrix}Y_1(s)\\ Y_2(s)\end{bmatrix}\begin{bmatrix}\dfrac{s+4}{s^2+4s+25} & \dfrac{s+5}{s^2+4s+25}\\ \dfrac{-25}{s^2+4s+25} & \dfrac{s-25}{s^2+4s+25}\end{bmatrix}\begin{bmatrix}U_1(s)\\ U_2(s)\end{bmatrix} \tag{6-14}$$

Por tanto, de (6-14), se obtiene

$$Y_1(s) = \frac{s+4}{s^2+4s+25}U_1(s)+\frac{s+5}{s^2+4s+25}U_2(s)$$

$$Y_2(s) = \frac{-25}{s^2+4s+25}U_1(s)+\frac{s-25}{s^2+4s+25}U_2(s)$$

Suponiendo $U_2(j\text{w}) = 0$ se encuentra $Y_1(j\text{w})/U_1(j\text{w})$ e $Y_2(j\text{w})/U_1(j\text{w})$ como sigue:

$$\frac{Y_1(j\omega)}{U_1(j\omega)} = \frac{j\omega+4}{(j\omega)^2+4j\omega+25}$$

$$\frac{Y_2(j\omega)}{U_1(j\omega)} = \frac{-25}{(j\omega)^2+4j\omega+25}$$

Análogamente, suponiendo $U_1(j\text{w}) = 0$ se encuentra $Y_1(j\text{w})/U_2(j\text{w})$ e $Y_2(j\text{w})/U_2(j\text{w})$ como sigue:

$$\frac{Y_1(j\omega)}{U_2(j\omega)} = \frac{j\omega+5}{(j\omega)^2+4j\omega+25}$$

$$\frac{Y_2(j\omega)}{U_2(j\omega)} = \frac{j\omega-25}{(j\omega)^2+4j\omega+25}$$

Observe que $Y_2(j\text{w})/U_2(j\text{w})$ es una función de transferencia de fase no mínima.

Para dibujar los diagramas de Bode de $Y_1(j\omega)/U_1(j\omega)$, $Y_2(j\omega)/U_1(j\omega)$, $Y_1(j\omega)/U_2(j\omega)$ e $Y_2(j\omega)/U_2(j\omega)$ con MATLAB, se puede utilizar la orden

```
bode(A,B,C,D)
```

LAB produce los diagramas de Bode cuando u_1 es la entrada y cuando u_2 es la entrada.Véase el programa de MATLAB 6-8 y los diagramas de Bode resultantes en la Figura 6-28.

Una manera alternativa de dibujar los cuatro diagramas individuales de Bode es utilizando la orden

```
bode(A,B,C,D,1)
```

Programa en MATLAB 6-8

```
A=[0 1;-25 -4]; B=[1 1;0 1]; C=[1 0;0 1]; D=[0 0;0 0];
bode(A,B,C,D)
title('Diagrama de bode')
%Pulsar cualquier tecla después de cada pantalla
```

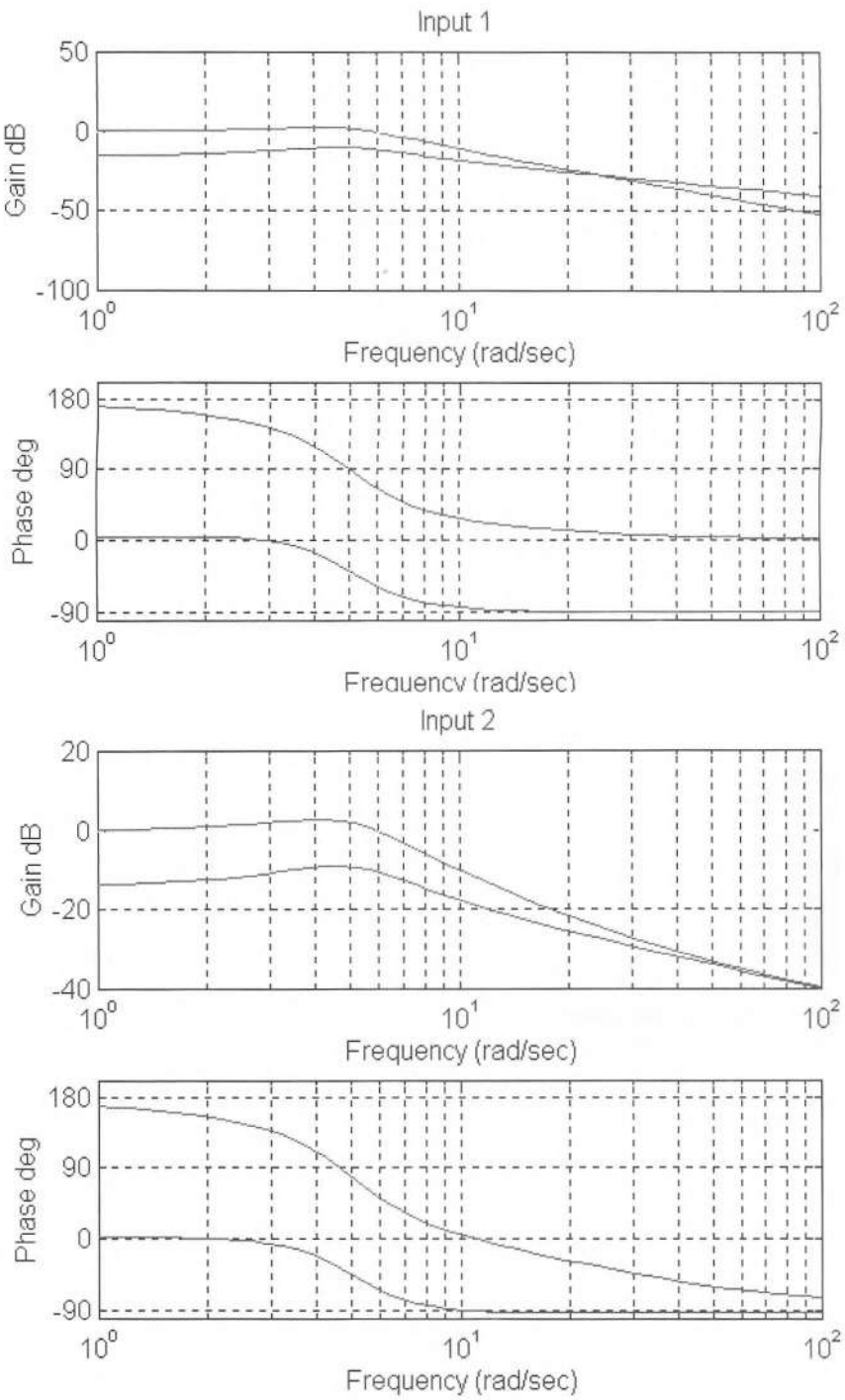


Figura 6-28

para obtener los diagramas de Bode de $Y_1(j\omega)/U_1(j\omega)$ y $Y_2(j\omega)/U_1(j\omega)$. Para obtener los diagramas de Bode de $Y_1(j\omega)/U_2(j\omega)$ y $Y_2(j\omega)/U_2(j\omega)$ utilice la siguiente orden:

```
bode(A,B,C,D,2)
```

Véase el programa de MATLAB 6-9. Los diagramas de Bode de este programa se muestran en las Figuras 6-29 y 6-30. En estos diagramas puede que no resulte fácil identificar qué curvas son de $Y_1(j\omega)$ y cuáles de $Y_2(j\omega)$. Lo lógico es utilizar la orden *text* para identificar las curvas. Sin embargo, la orden *text* no ha sido aplicada a las órdenes de *bode* presentes.

Para utilizar la orden *text*, necesitamos utilizar la siguiente orden:

```
[mag,fase,w] = bode(A,B,C,D,iu,w)
```

Las matrices mag y fase contienen las magnitudes y los ángulos de fase de $Y_1(j\omega)$ e $Y_2(j\omega)$ evaluados en cada punto de frecuencia considerado.

Para dibujar la magnitud de $Y_1(j\mathrm{w})$, primero especifique el rango de frecuencias introduciendo, por ejemplo,

```
w = logspace(-1,2,100)
```

en el computador. (Esto significa que el rango de frecuencias va desde $\mathrm{w} = 10^{-1} = 0.1$ hasta $\mathrm{w} = 10^{2} = 100$ y calcule 100 puntos espaciados logarítmicamente de una forma uniforme). Después utilice la siguiente orden:

```
Y1 = mag*[1;0]
```

Programa en MATLAB 6-9

```
%-------Diagrama de Bode-------

%*****En este programa se obtendrán los diagramas de Bode de un sistema con dos
%entradas, u1 y u2*****

%*****Introduzca la matriz de estado A, la matriz de control B, la matriz de
%salida C y la matriz de transmisión directa D*****

A=[0 1;-25 -4];
B=[1 1;0 1];
C=[1 0;0 1];
D=[0 0;0 0];

%*****Primero, se obtendrá la respuesta en frecuencia cuando u2=0; es decir, se
%obtiene Y1(jw)/U1(jw) y Y2(jw)/U1(jw)*****

%*****Introduzca la orden Bode(A,B,C,D,1)*****

bode(A,B,C,D,1), title('Diagramas de Bode: bode(A,B,C,D,1)')

%*****A continuación, se obtendrá la respuesta en frecuencia cuando u1=0; es
%decir, se obtiene Y1(jw)/U2(jw) y Y2(jw)/U2(jw)*****

%*****Introduzca la orden bode(A,B,C,D,2)*****

bode(A,B,C,D,2), title('Diagramas de Bode: Bode(A,B,C,D,2)')
```

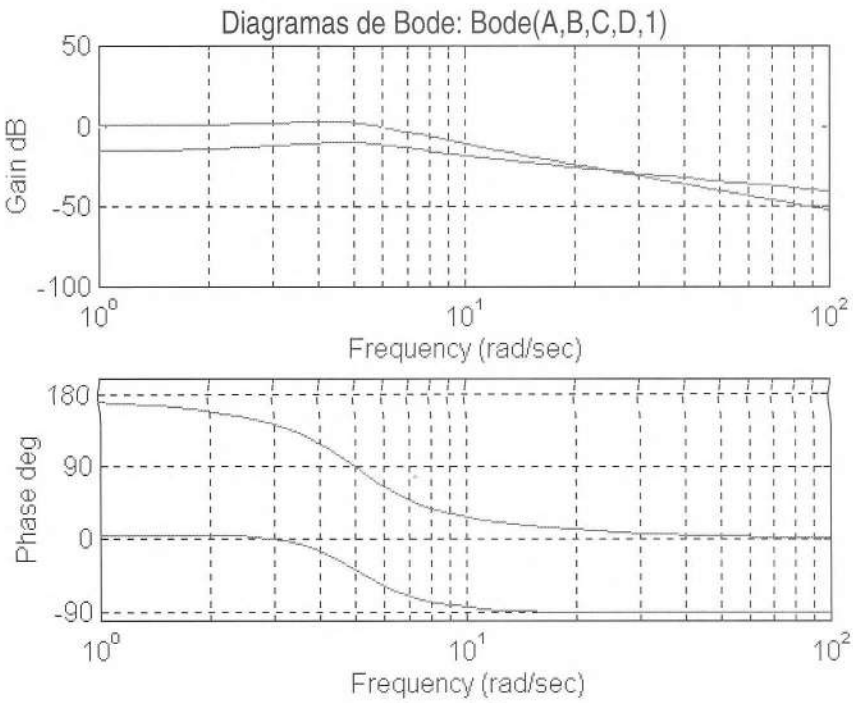


Figura 6-29

Para convertir la magnitud a decibelios, utilice la declaración

```
magdB = 20*log10(mag)
```

Por tanto, para convertir Y1 a decibelios, introduzca la declaración

```
Y1dB = 20*log10(Y1)
```

Finalmente, introduzca la orden

```
semilogx(w,Y1dB,'o',w,Y1dB,'-')
```

o

```
semilogx(w,Y1dB,'o')
```

Análogamente, para dibujar la magnitud de Y2 en decibelios, utilice las siguientes órdenes:

```
Y2 = mag*[0;1];
Y2dB = 20*log10(Y2);
semilogx(w,Y2dB,'x',w,Y2dB,'-')
```

La orden *text* se puede utilizar para dibujar texto en la figura.
Análogamente para dibujar los ángulos de fase para $Y_1(j\omega)$ e $Y_2(j\omega)$, utilice las órdenes

```
Y1p = fase*[1;0];
semilogx(w,Y1p,'o',w,Y1p,'-')
```

e

```
Y2p = fase*[0;1];
semilogx(w,Y2p,'o',w,Y2p,'-')
```

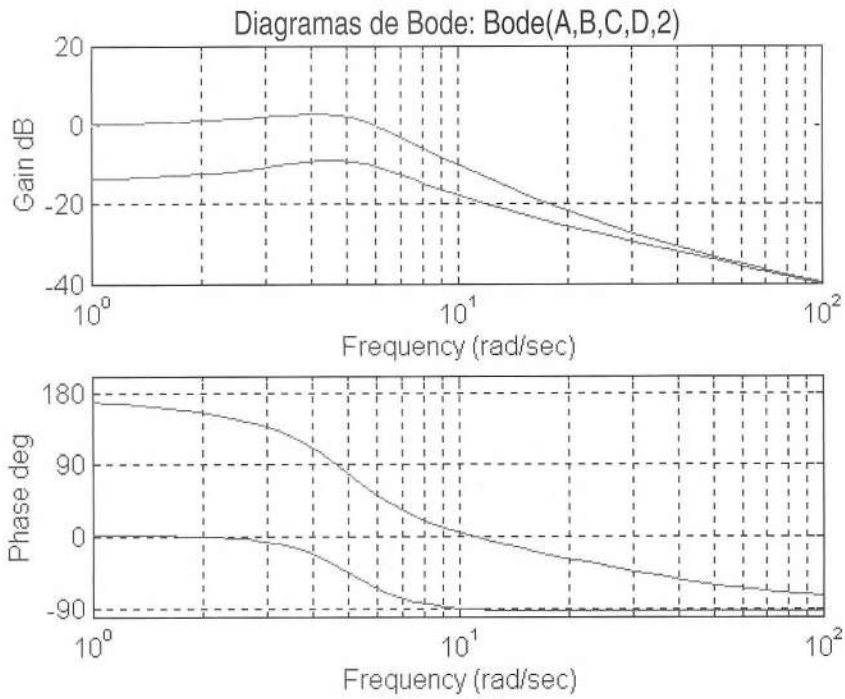


Figura 6-30

Si desea dibujar dos curvas de fase en un único diagrama, se necesita especificar un rango de ángulos de fase adecuado. Una forma de hacer esto es dibujar líneas no visibles en 220° y -100° introduciendo las órdenes y declaraciones siguientes:

```
fmax = 220*ones(1,100);
fmin = -100*ones(1,100);
semilogx(w,Y1p,'o',w,Y1p,'-',w,fmax,'--i',w,fmin,':i')
```

(Observe que el número de puntos de fmax y el número de puntos de fmin debe de coincidir con el número de puntos de frecuencias, el cual en este caso es 100). La línea fmax y la línea fmin no son visibles ya que se utilizan las marcas '--i' y ':i'.

Como se expuso con anterioridad, la técnica descrita arriba se puede utilizar también para el dibujo de las curvas de magnitud. Si se necesita que el rango de magnitud se encuentre al menos entre -100 dB y +40 dB, debe de introducir líneas no visibles en -100 dB y en +40 dB en el dibujo, especificando

```
dBmax = 40*ones(1,100)
dBmin = -100*ones(1,100)
```

e introduciendo la orden

```
semilogx(w,Y1dB,'o',w,Y1dB,'-',w,dBmax,'--i',w,dBmin,':i')
```

El programa de MATLAB 6-10 dibuja los diagramas de Bode de $Y_1(j\omega)/U_1(j\omega)$, $Y_2(j\omega)/U_1(j\omega)$, $Y_1(j\omega)/U_2(j\omega)$ y $Y_2(j\omega)/U_2(j\omega)$ con un rango de frecuencia y un rango de ángulos de fase especificados por el usuario. Los diagramas de Bode resultantes se muestran en las Figuras 6-31 y 6-32.

Observe que el efecto de añadir la línea no visible en -100°, por ejemplo, puede destacar en la copia dura sobre la abcisa en -100°. Si no quiere este efecto, cambie la línea de -100° a -95°, por ejemplo, la línea en -95° no destacará en la copia dura. El efecto de las líneas no visibles aparece únicamente si una línea no visible coincide con el borde superior o inferior de abcisas.

Programa en MATLAB 6-10

```
%-------Diagrama de Bode-------

%*****En este programa se obtendran los diagramas de Bode de un sistema de dos
%entradas (u1 y u2) y dos salidas (y1 e y2)*****

%*****Introduzca la matriz de estado A, la matriz de control B, la matriz de
%salida C y la matriz de transmisión directa D*****

A=[0 1;-25 -4]; B=[1 1;0 1];
C=[1 0;0 1]; D=[0 0;0 0];

%*****Primero, se obtendrá la respuesta en frecuencia cuando u2=0; es decir, se
%obtiene Y1(jw)/U1(jw) y Y2(jw)/U1(jw)*****

%*****Especifique el rango de frecuencias e introduzca la orden
%[mag1,fase1,w]=bode(A,B,C,D,1,w)*****

w=logspace(-1,3,100);
[mag1,fase1,w]=bode(A,B,C,D,1,w);

%*****Para representar los diagramas de bode convierta las magnitudes Y1 e Y2 a
%decibelios*****

Y1=mag1*[1;0];Y1dB=20*log10(Y1);
Y2=mag1*[0;1];Y2dB=20*log10(Y2);

%*****Para representar los diagramas de bode para Y1(jw)/U1(jw) y Y2(jw)/U1(jw)
%en un único gráfico utilice la orden hold, se representa el segundo diagrama en
%primer lugar porque su rango de magnitud es mayor que la del primero*****

semilogx(w,Y2dB,'x',w,Y2dB,'-')
grid, title('Diagramas de bode: Entrada=u1(u2=0)')
xlabel('Frecuencia (rad/seg)'), ylabel('Ganancia dB')
text(1,2,'Y2')
hold

%Current plot held

semilogx(w,Y1dB,'o',w,Y1dB,'-')
text(1,-14,'Y1')
hold

%Current plot released

%*****A continuación, se representan las curvas de fase. Un posible rango para
%la fase de las dos curvas es el comprendido entre -100 grados y +220 grados.
%Introduzca fmax y fmin en el programa y dibuje las líneas fmax y fmin en un
%color no visible. Para representarlas curvas de fase y las líneas no visibles,
%introduzca lo siguiente: fmax, fmin, Y1p, Y2p y la orden semilogx*****

fmax=220*ones(1,100);
fmin=-100*ones(1,100);
Y1p=fase1*[1;0];
```

```
Y2p=fase1*[0;1];
semilogx(w,Y1p,'o',w,Y1p,'-',w,fmax,'--i',w,fmin,':i')
grid
xlabel('Frecuencia (rad/seg)'), ylabel('Fase grados')
text(1,10,'Y1')
hold

%Current plot held

semilogx(w,Y2p,'x',w,Y2p,'-')
text(1,172,'Y2')
hold

%Current plot released

%*****A continuaciÛn, se obtendrá la respuesta en frecuencia cuando u1=0;
%es decir se obtiene Y1(jw)/U2(jw) y Y2(jw)/U2(jw)*****

[mag2,fase2,w]=bode(A,B,C,D,2,w);
YY1=mag2*[1;0];YY1dB=20*log10(YY1);
YY2=mag2*[0;1];YY2dB=20*log10(YY2);
semilogx(w,YY2dB,'x',w,YY2dB,'-')
grid, title('Diagramas de bode: Entrada=u2(u1=0)')
xlabel('Frecuencia (rad/seg)'), ylabel('Ganancia dB')
text(1,2,'Y2'); hold

%Current plot held

semilogx(w,YY1dB,'o',w,YY1dB,'-')
text(1,-12,'Y1')
hold

%Current plot released

%*****A continuaciÛn, se representan las curvas de fase. Un posible
%rango para la fase de las dos curvas es el comprendido entre -100 grados
%y +220 grados. Se utiliza los mismos fmax y fmin que se han definido
%anteriormente.Introduzca YY1p y YY2p como se define a continuaciÛn. Después
%introduzca la orden semilogx*****

YY1p=fase2*[1;0];
YY2p=fase2*[0;1];
semilogx(w,YY1p,'o',w,YY1p,'-',w,fmax,'--i',w,fmin,':i')
grid
xlabel('Frecuencia (rad/seg)'),ylabel('Fase grados')
text(1,7,'Y1')
hold

%Current plot held

semilogx(w,YY2p,'x',w,YY2p,'-')
text(1,170,'Y2')
hold

%Current plot released
```

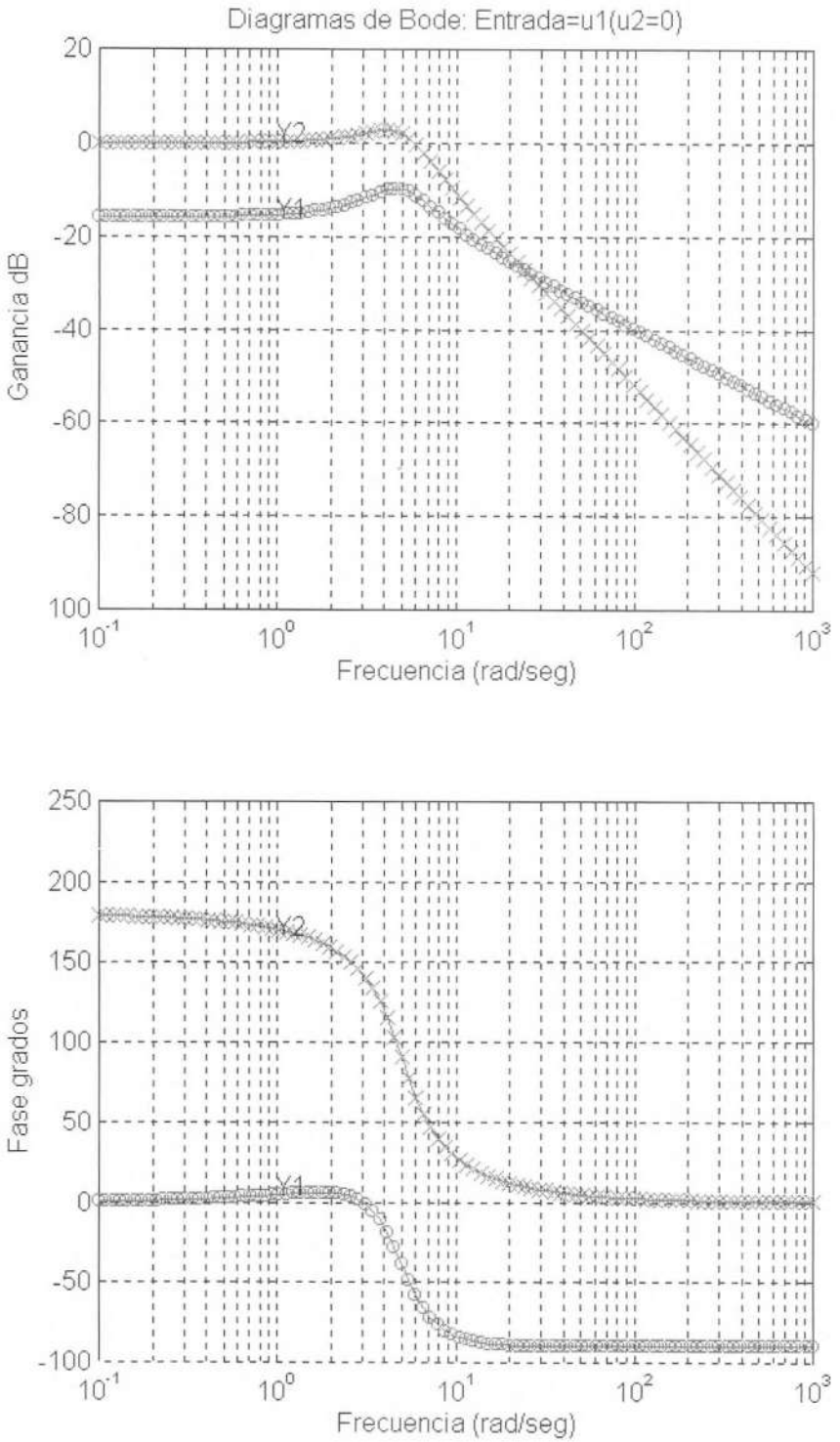

Diagramas de Bode: Entrada=u1(u2=0)
Ganancia dB
Fase grados
Frecuencia (rad/seg)
Frecuencia (rad/seg)
Y1
Y2
Y1
Y2

Figura 6-31

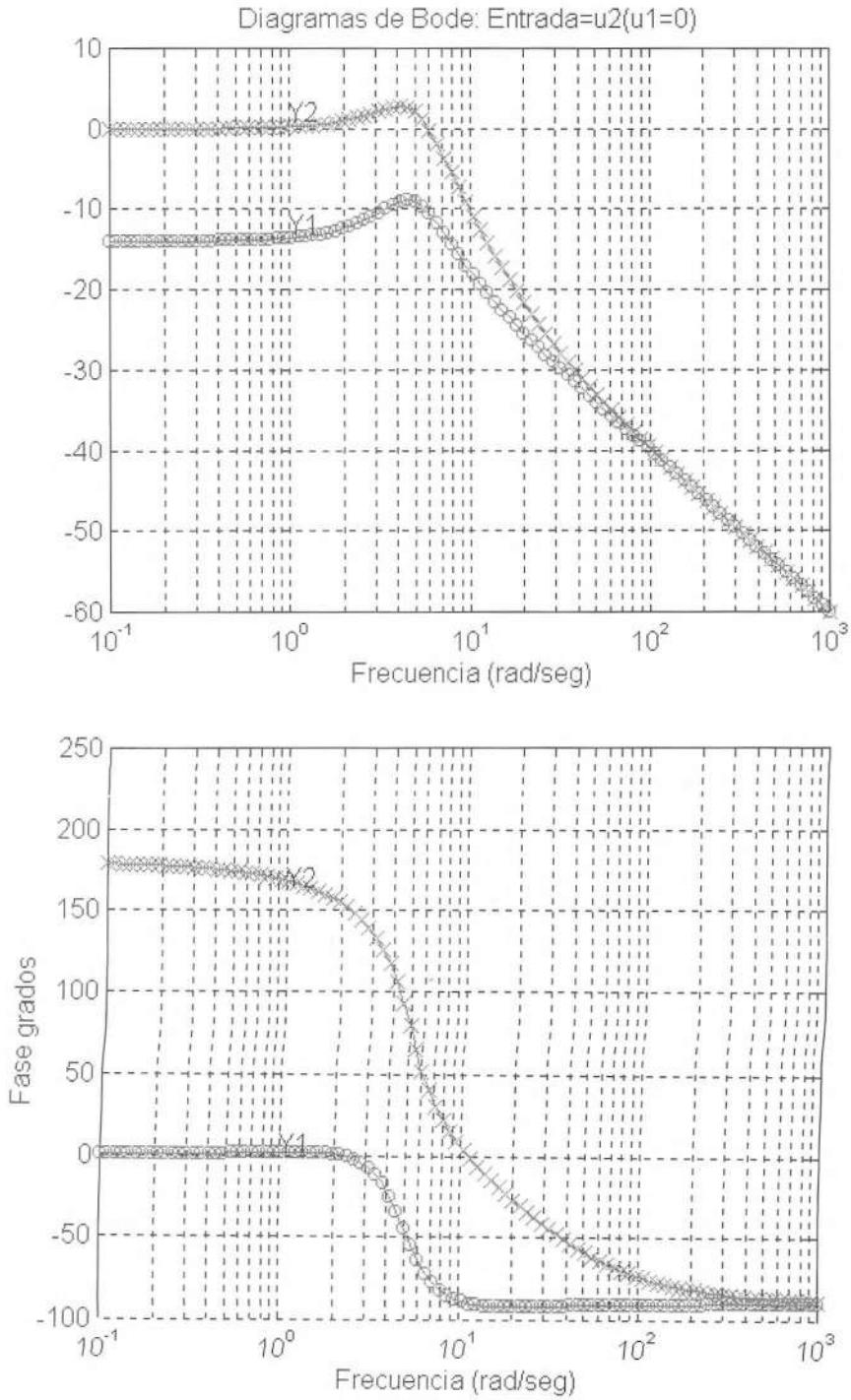


Figura 6-32

6.4 OBTENCIÓN DEL DIAGRAMA DE NYQUIST CON MATLAB

Los diagramas de Nyquist, al igual que los diagramas de Bode, son utilizados normalmente para la representación de la respuesta en frecuencia de sistemas de control realimentados, lineales e invariantes en el tiempo.

Los diagramas de Nyquist son diagramas polares, mientras que los diagramas de Bode son gráficos rectangulares. Cada uno de los gráficos puede ser más adecuado para una operación en particular, pero dada una operación, ésta siempre se puede llevar a un gráfico o a otro.

La orden nyquist calcula la respuesta en frecuencia para sistemas continuos, lineales e invatiantes en el tiempo. Cuando se llama sin argumentos en el lado izquierdo, nyquist produce el diagrama de Nyquist en la pantalla, nyquist se puede utilizar convenientemente para determinar la estabilidad de un sistema con realimenta-ción unidad basado en el criterio de estabilidad de Nyquist. El sistema en lazo cerrado será estable si el diagrama de Nyquist encierra al punto $-1 + j0$ exactamente P veces en la dirección contraria a las agujas del reloj, donde P es el número de polos en lazo abierto en el semiplano derecho del plano s.

La orden

```
nyquist(num,den)
```

dibuja el diagrama de Nyquist de la función de transferencia

$$G(s) = \frac{\text{den}(s)}{\text{num}(s)}$$

donde num y den contiene los coeficientes polinomiales en potencias descendentes de s.

La orden

```
nyquist(num,den,w)
```

utiliza el vector de frecuencias especificado por el usuario w. El vector w especifica los puntos de frecuencia en radianes por segundo en los cuales se va a calcular la respuesta en frecuencia.

Cuando se llama con argumentos en el lado izquierdo

```
[re,im,w] = nyquist(num,den)
```

o

```
[re,im,w] = nyquist(num,den,w)
```

MATLAB devuelve la respuesta en frecuencia del sistema en las matrices re, im y w. Ningún gráfico se dibuja en pantalla. Las matrices re e im contienen las partes reales e imaginarias de la respuesta en frecuencia del sistema evaluado en los puntos de frecuencia especificado en el vector w. Observe que re e im tienen tantas columnas como salidas y una fila por cada elemento en w.

EJEMPLO 6-13

Considere la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{1}{s^2 + 0{,}8s + 1}$$

Dibuje un diagrama de Nyquist con MATLAB.

Como el sistema viene dado por su función de transferencia, la orden

```
nyquist(num,den)
```

se debe de utilizar para dibujar el diagrama de Nyquist. El programa de MATLAB 6-11 produce el diagrama de Nyquist que se muestra en la Figura 6-33. En este gráfico los rangos para el eje real y el eje imaginario se determinan automáticamente.

Los diagramas de Nyquist son diagramas polares, mientras que los diagramas de Bode son gráficos rectangulares. Cada uno de los gráficos puede ser más adecuado para una operación en particular, pero dada una operación, ésta siempre se puede llevar a un gráfico o a otro.

La orden nyquist calcula la respuesta en frecuencia para sistemas continuos, lineales e invatiantes en el tiempo. Cuando se llama sin argumentos en el lado izquierdo, nyquist produce el diagrama de Nyquist en la pantalla, nyquist se puede utilizar convenientemente para determinar la estabilidad de un sistema con realimenta-ción unidad basado en el criterio de estabilidad de Nyquist. El sistema en lazo cerrado será estable si el diagrama de Nyquist encierra al punto $-1 + j0$ exactamente P veces en la dirección contraria a las agujas del reloj, donde P es el número de polos en lazo abierto en el semiplano derecho del plano s.

La orden

```
nyquist(num,den)
```

dibuja el diagrama de Nyquist de la función de transferencia

$$G(s) = \frac{\text{den}(s)}{\text{num}(s)}$$

donde num y den contiene los coeficientes polinomiales en potencias descendentes de s.

La orden

```
nyquist(num,den,w)
```

utiliza el vector de frecuencias especificado por el usuario w. El vector w especifica los puntos de frecuencia en radianes por segundo en los cuales se va a calcular la respuesta en frecuencia.

Cuando se llama con argumentos en el lado izquierdo

```
[re,im,w] = nyquist(num,den)
```

o

```
[re,im,w] = nyquist(num,den,w)
```

MATLAB devuelve la respuesta en frecuencia del sistema en las matrices re, im y w. Ningún gráfico se dibuja en pantalla. Las matrices re e im contienen las partes reales e imaginarias de la respuesta en frecuencia del sistema evaluado en los puntos de frecuencia especificado en el vector w. Observe que re e im tienen tantas columnas como salidas y una fila por cada elemento en w.

EJEMPLO 6-13

Considere la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{1}{s^2 + 0{,}8s + 1}$$

Dibuje un diagrama de Nyquist con MATLAB.

Como el sistema viene dado por su función de transferencia, la orden

```
nyquist(num,den)
```

se debe de utilizar para dibujar el diagrama de Nyquist. El programa de MATLAB 6-11 produce el diagrama de Nyquist que se muestra en la Figura 6-33. En este gráfico los rangos para el eje real y el eje imaginario se determinan automáticamente.

Programa en MATLAB 6-11

```
num=[0 0 1];
den=[1 0.8 1];
nyquist(num,den)
grid
title('Diagrama de Nyquist de G(s)=1/(s^2+0.8s+1)')
```

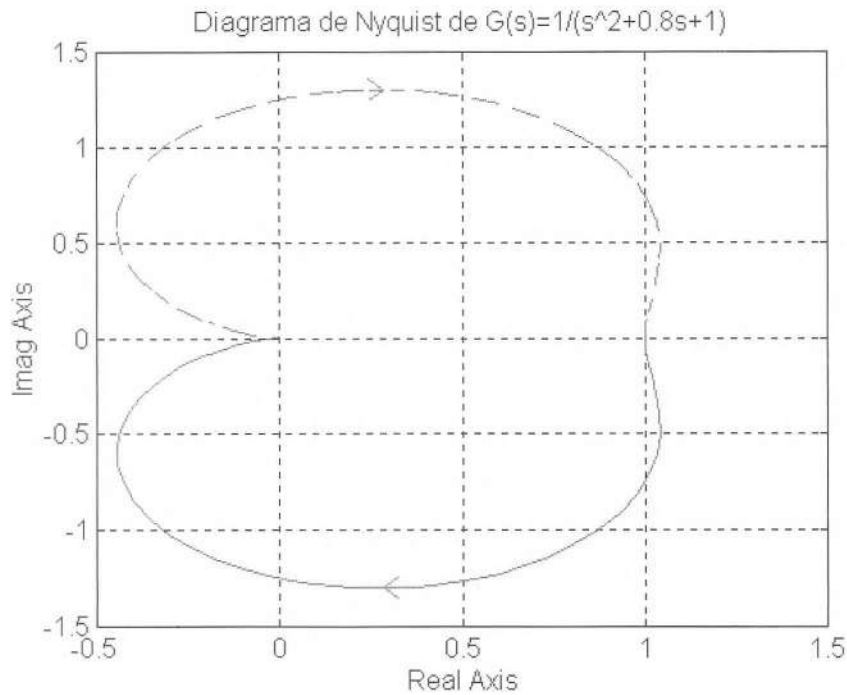


Figura 6-33

Si se desea dibujar el diagrama de Nyquist determinando manualmente los rangos, por ejemplo desde -2 hasta 2 en el eje real y desde -2 hasta 2 en el eje imaginario, introduzca la siguiente orden en el computador:

```
v = [-2  2  -2  2];
axis(v);
```

o, junte estas dos líneas en una,

```
axis([-2  2  -2  2]);
```

Véase el programa de MATLAB 6-12, el diagrama de Nyquist resultante se muestra en la Figura 6-34.

Programa en MATLAB 6-12

```
num=[0 0 1]; den=[1 0.8 1];
v=[-2 2 -2 2];
axis(v);
nyquist(num,den)
grid, title('Diagrama de Nyquist de G(s)=1/(s^2+0.8s+1)')
```

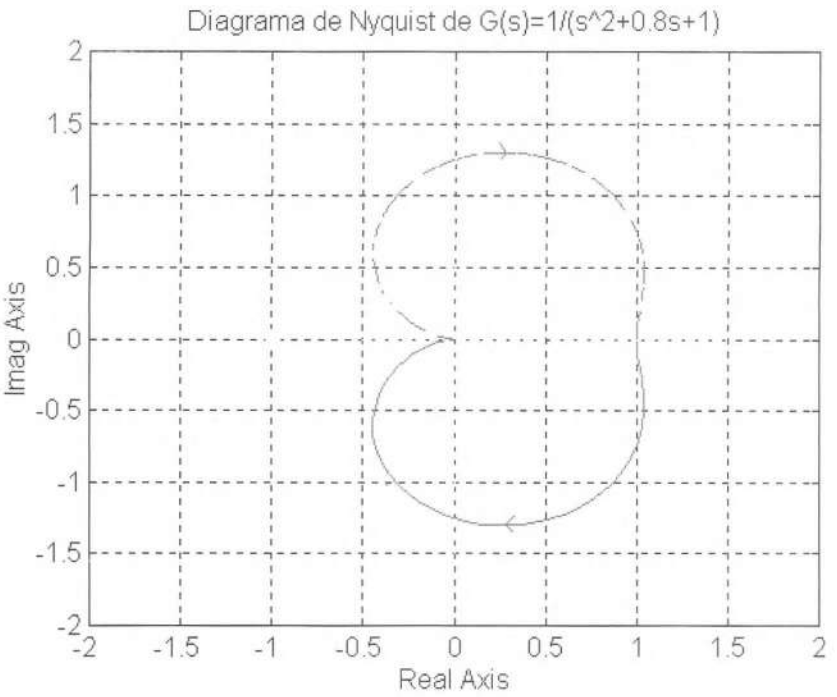


Figura 6-34

EJEMPLO 6-14

Dibuje un diagrama de Nyquist para la siguiente $G(s)$:

$$G(s) = \frac{1}{s(s+1)}$$

Si se introduce el programa de MATLAB 6-13 en el computador, aparece el siguiente mensaje de aviso:

'Divide by zero'

Programa en MATLAB 6-13

```
num=[0 0 1];
den=[1 1 0];
nyquist(num,den)

%Warning:Divide by zero

title('Diagrama de Nyquist incorrecto')
```

Esto corresponde a un caso singular en los cálculos de MATLAB. Si se utiliza el programa de MATLAB 6-13, el diagrama de Nyquist resultante mostrado en la Figura 6-35 es incorrecto. Este diagrama de Nyquist se pude corregir si se especifica el axis(v). Por ejemplo, si se introduce la orden

```
v = [-2  2  -5  5];
axis(v);
```

en el computador, como se observa en el programa de MATLAB 6-14, se obtiene un diagrama de Nyquist correcto como se muestra en la Figura 6-36.

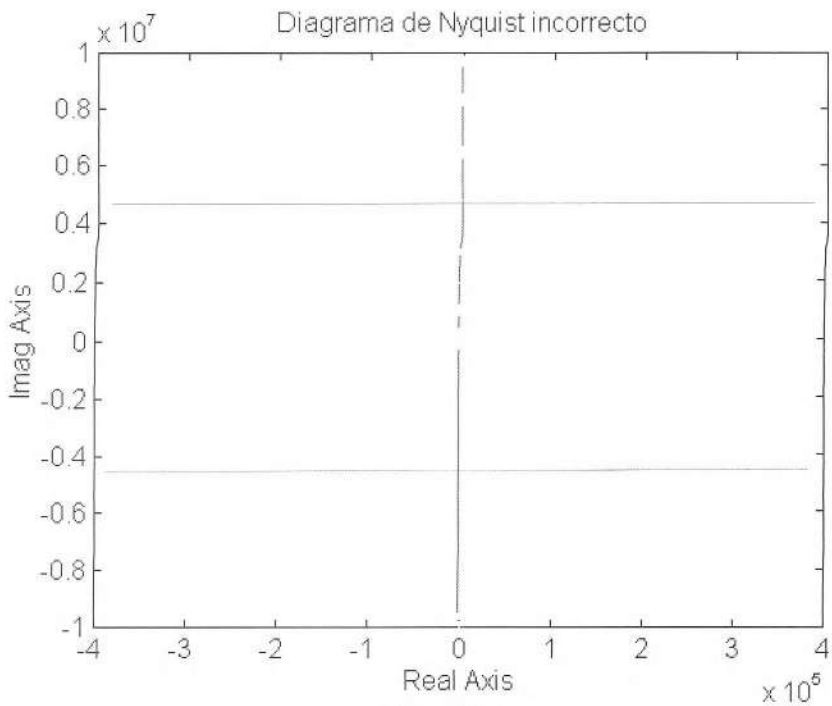


Figura 6-35

Programa en MATLAB 6-14

```
num=[0 0 1];
den=[1 1 0];
v=[-2 2 -5 5];
axis(v);
nyquist(num,den)

%Warning: Divide by zero

grid
title('Diagrama de Nyquist de G(s)=1/[s(s+1)]')
```

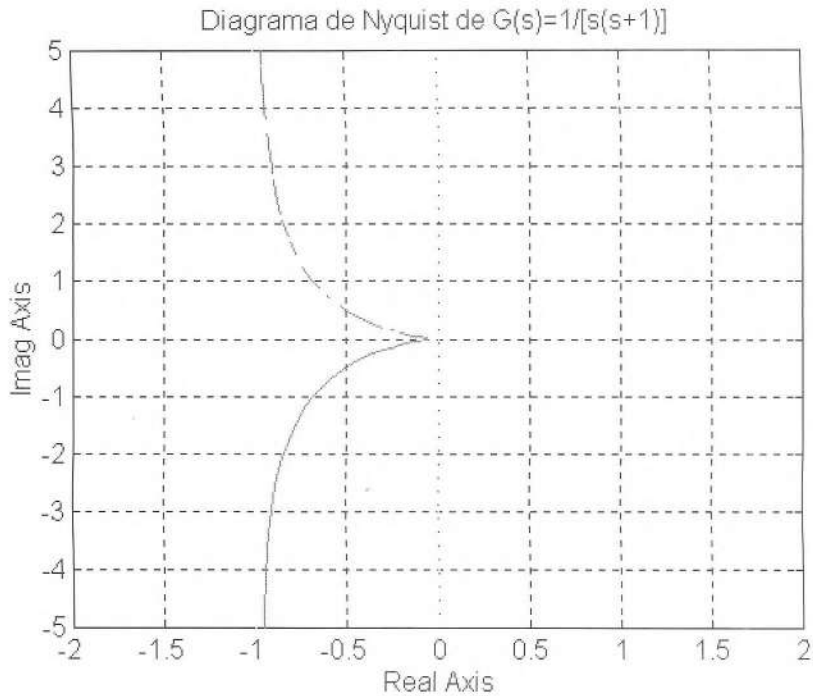


Figura 6-36

Observe que el diagrama de Nyquist que se da en la Figura 6-36 incluye el lugar para $w > 0$ y para $w < 0$. Si se desea dibujar únicamente el diagrama de Nyquist para la región de frecuencias positivas ($w > 0$), se necesita utilizar la orden

[re,im,w] = nyquist(num,den,w)

Un programa de MATLAB utilizando esta orden *nyquist* se muestra en el programa de MATLAB 6-15. El diagrama de Nyquist resultante se presenta en la Figura 6-37.

Programa en MATLAB 6-15

```
num=[0 0 1];
den=[1 1 0];
w=0.1:0.1:100;
[re,im,w]=nyquist(num,den,w);
v=[-2 2 -5 5];
axis(v);
plot(re,im)
grid
title('Diagrama de Nyquist de G(s)=1/[s(s+1)]')
xlabel('Eje real')
ylabel('eje imaginario')
```

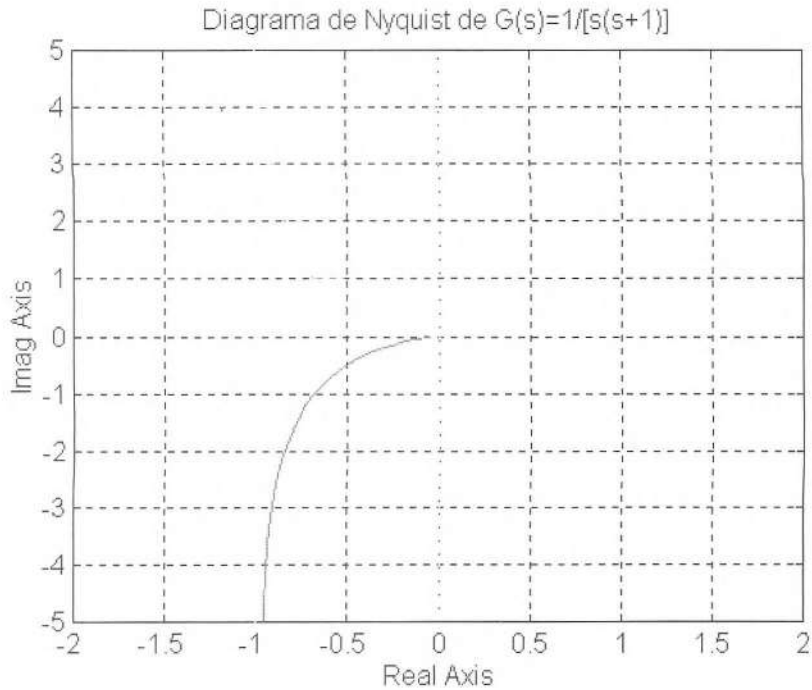


Figura 6-37

EJEMPLO 6-15

Considere un sistema con realimentación unitaria con la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{s^2 + 2s + 1}{s^3 + 0{,}2s^2 + s + 1}$$

Dibuje un diagrama de Nyquist y examine la estabilidad en lazo cerrado.

El programa de MATLAB 6-16 dibuja un diagrama de Nyquist para la $G(s)$ dada. El gráfico resultante se muestra en la Figura 6-38.

Observe que el punto $-1+j0$ es encerrado dos veces en el sentido contrario al de las agujas del reloj. Para examinar la localización de los tres polos de $G(s)$, utilice el siguiente programa de MATLAB:

```
den=[1 0.2 1 1];
roots(den)

ans =

   0.2623 + 1.1451i
   0.2623 - 1.1451i
  -0.7246
```

Claramente hay dos polos en el semiplano derecho. Por tanto, en el criterio de estabilidad de Nyquist $P = 2$. Como el diagrama de Nyquist de G(s) encierra al punto $-1+j0$ dos veces en el sentido contrario al de las agujas del reloj, el sistema en lazo cerrado es estable

Programa en MATLAB 6-16

```
num=[0 1 2 1];
den=[1 0.2 1 1];
nyquist(num,den)
grid
title('Diagrama de Nyquist de G(s)=(s^2+2s+1)/(s^3+0.2s^2+s+1)')
```

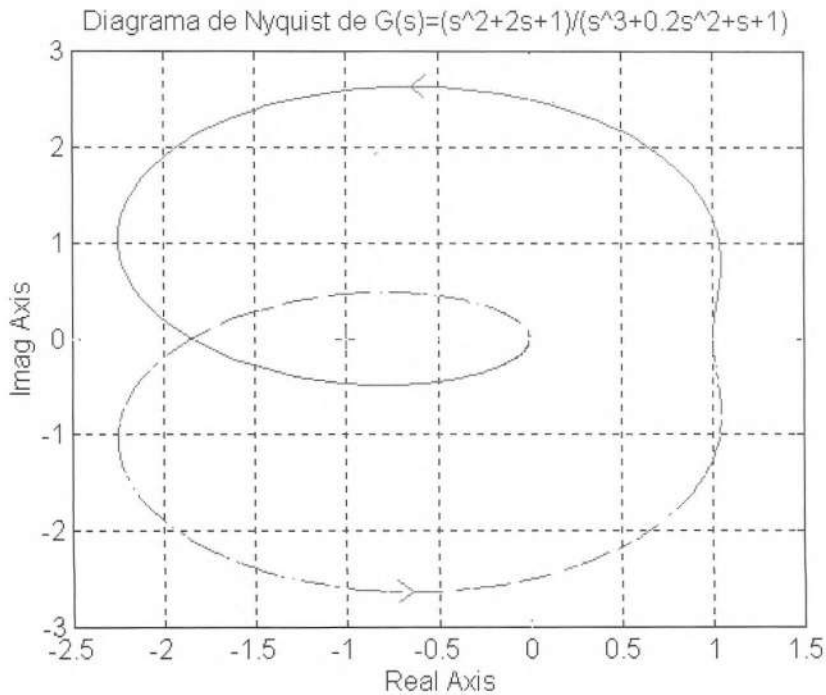


Figura 6-38

EJEMPLO 6-16

Considere el sistema con realimentación unitaria con la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{1}{s^3 + 0{,}2s^2 + s + 1}$$

Dibuje un diagrama de Nyquist y examine la estabilidad del sistema.

Un programa de MATLAB para dibujar el diagrama de Nyquist para esta $G(s)$ se muestra en el programa de MATLAB 6-17. El diagrama de Nyquist resultante se muestra en la Figura 6-39.

Para dibujar el diagrama de Nyquist en un dominio mayor, es decir, el eje real desde -2 hasta 2 y el eje imaginario desde -1 hasta 1, se puede introducir el programa de MATLAB 6-18 en el computador. El gráfico resultante se muestra en la Figura 6-40.

Programa en MATLAB 6-17

```
num=[0 0 0 1];
den=[1 0.2 1 1];
nyquist(num,den)
grid
title('Diagrama de Nyquist de G(s)=1/(s^3+0.2s^2+s+1)')
```

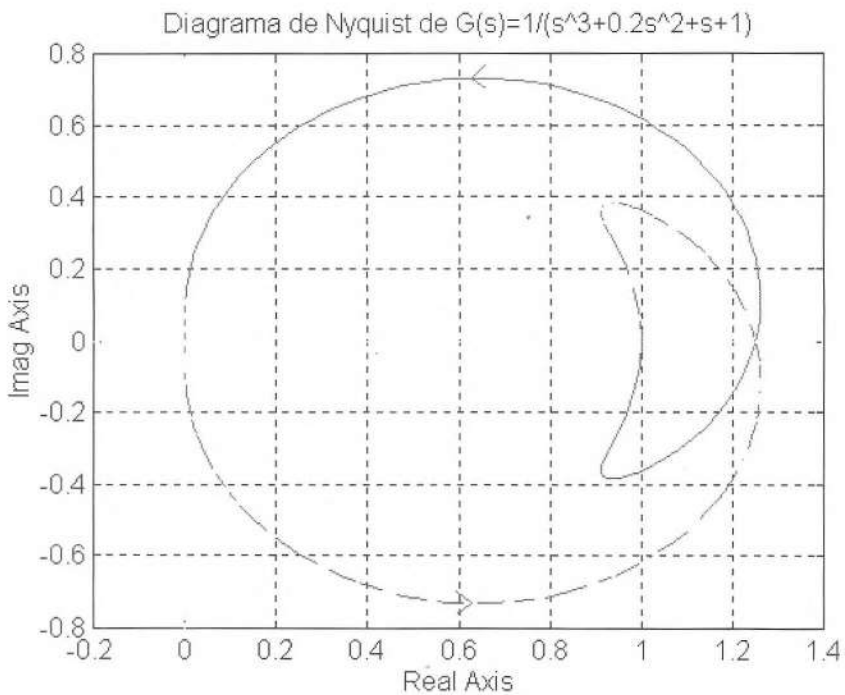


Figura 6-39

Programa en MATLAB 6-18

```
num=[0 0 0 1];
den=[1 0.2 1 1];
v=[-2 2 -1 1];
axis(v);
nyquist(num,den)
grid
title('Diagrama de Nyquist de G(s)=1/(s^3+0.2s^2+s+1)')
```

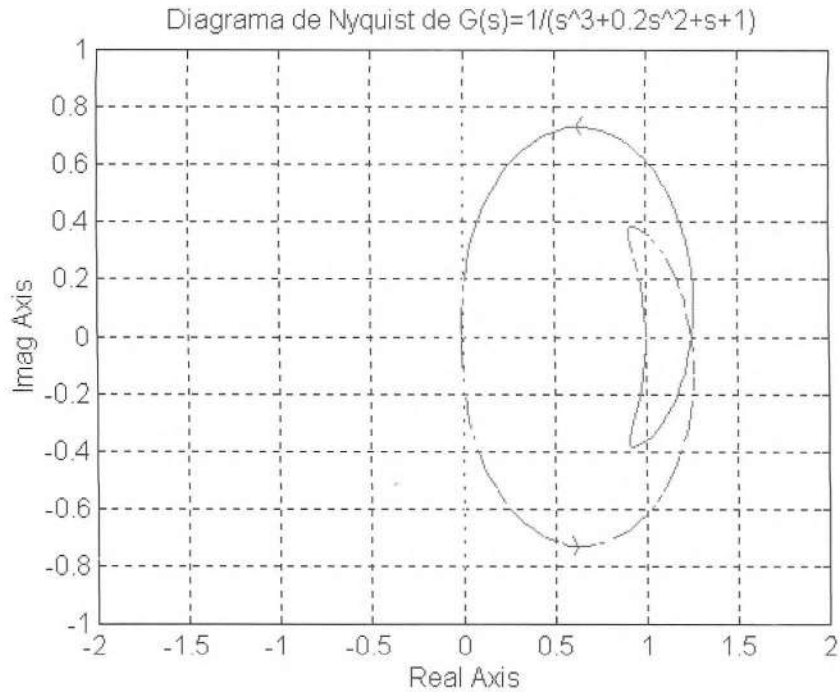


Figura 6-40

De la Figura 6-39 (o 6-40) se ve que el diagrama de Nyquist no encierra al punto -1+*j*0. Si $G(s)$ no tenía polos en el semiplano derecho, el sistema en lazo cerrado podría ser estable. Sin embargo, $G(s)$ tiene dos polos en el semiplano derecho del plano s ($s = 0.2623 \pm j1.1451$). Para la estabilidad, el diagrama de Nyquist debe de encerrar al punto 1+*j*0 dos veces en el sentido contrario al de las agujas del reloj. Como el diagrama de Nyquist no encierra al punto 1+*j*0 el sistema en lazo cerrado es inestable.

EJEMPLO 6-17

Considere un sistema con realimentación unidad y la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{20(s^2 + s + 0{,}5)}{s(s+1)(s+10)}$$

Dibuje un diagrama de Nyquist con MATLAB y examine la estabilidad del sistema en lazo cerrado.

Primero introduzca el programa de MATLAB 6-19 en el computador. La pantalla mostrará el mensaje de aviso "Divide by zero" y no dará el diagrama de Nyquist correcto. Éste es un caso singular en los cálculos de MATLAB. La Figura 6-41 muestra el diagrama incorrecto generado por el programa de MATLAB 6-19.

Este diagrama de Nyquist incorrecto se puede corregir introduciendo la orden *axis*, como se muestra en el programa de MATLAB 6-20. El diagrama de Nyquist resultante se muestra en la Figura 6-42. Como no hay polos del lazo abierto en el semiplano derecho, $P = 0$ en el criterio de estabilidad de Nyquist. En la Figura 6-42 se ve que el diagrama de Nyquist no encierra al punto -1+*j*0. Por tanto el sistema en lazo cerrado es estable.

Programa en MATLAB 6-19

```
num=[0 20 20 10];
den=[1 11 10 0];
nyquist(num,den)

%Warning: Divide by zero

title('Diagrama de Nyquist incorrecto')
```

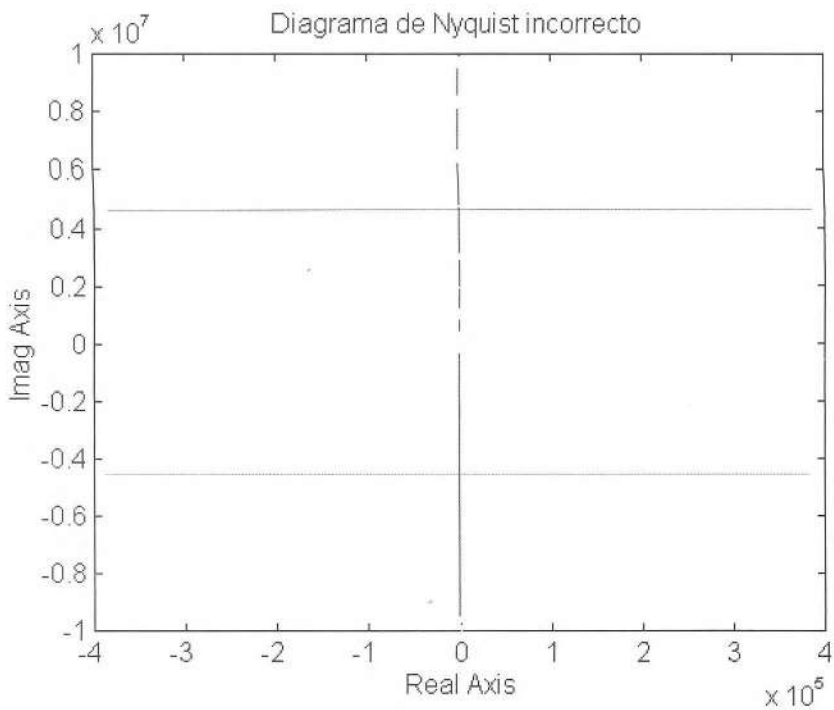


Figura 6-41

Programa en MATLAB 6-20

```
num=[0 20 20 10];
den=[1 11 10 0];
v=[-2 3 -3 3];
axis(v);
nyquist(num,den)

%Warning: Divide by zero

grid
title('Diagrama de Nyquist de G(s)=20(s^2+s+0.5)/[s(s+1)(s+10)]')
```

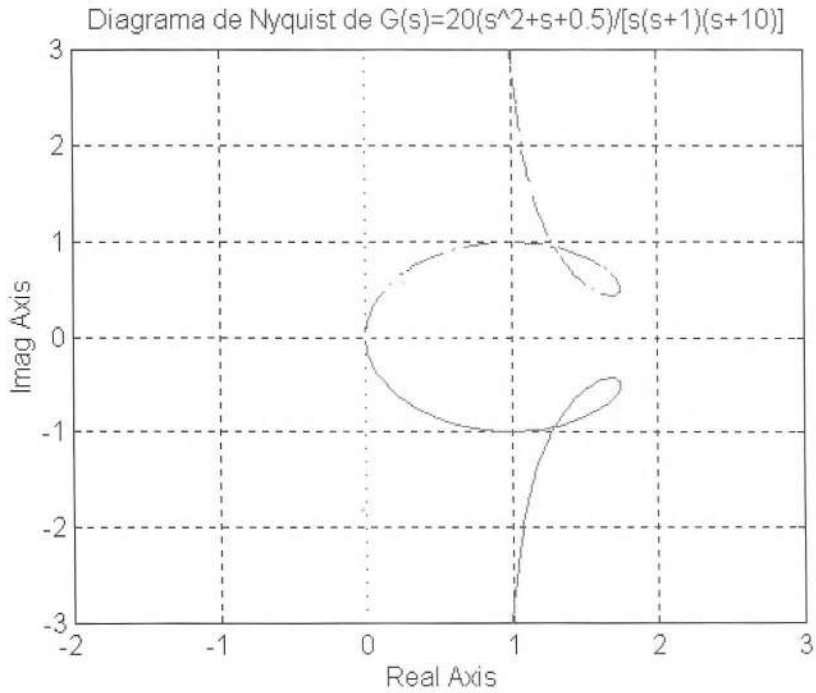


Figura 6-42

A continuación se va a dibujar el diagrama de Nyquist únicamente para la región de frecuencias positivas. Esto se puede realizar utilizando la orden

```
[re,im,w] = nyquist(num,den,w)
```

La región de frecuencia se puede dividir en varias sub-regiones utilizando diferentes incrementos. Por ejemplo, la región de frecuencia de interés se puede dividir en tres sub-regiones como sigue:

```
w1 = 0.1:0.1:10;
w2 = 10:2:100;
w3 = 100:10:500;
w = [w1  w2  w3]
```

El programa de MATLAB 6-21 utiliza esta región de frecuencia. Utilice este programa para obtener el diagrama de Nyquist de la Figura 6-43.

Programa en MATLAB 6-21

```
num=[0 20 20 10];
den=[1 11 10 0];
w1=0.1:0.1:10;w2=10:2:100;w3=100:10:500;
w=[w1 w2 w3];
[re,im,w]=nyquist(num,den,w);
v=[-3 3 -5 1];
axis(v);
plot(re,im)
grid, title('Diagrama de Nyquist de G(s)=20(s^2+s+0.5)/[s(s+1)(s+1  ]')
xlabel('Eje real')
ylabel('Eje imaginario')
```

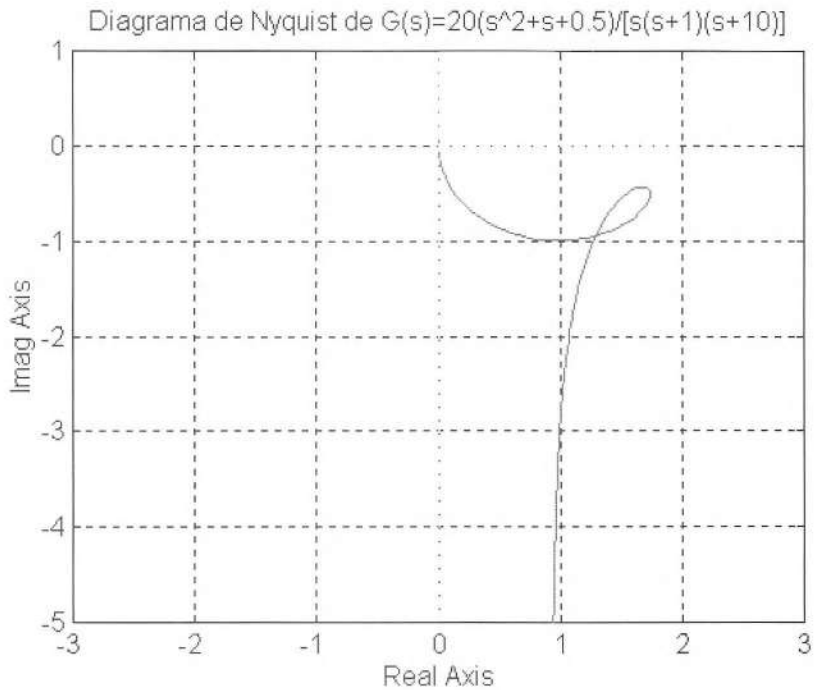


Figura 6-43

EJEMPLO 6-18

Considere el sistema de realimentación negativa con la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{s^2 + 4s + 6}{s^2 + 5s + 4}$$

Dibuje un diagrama de Nyquist.

El programa de MATLAB 6-22 dará el diagrama de Nyquist, el cual se muestra en la Figura 6-44. Este sistema es estable, ya que no hay polos en lazo abierto en el semiplano derecho y no se encierra al punto $-1+j0$.

Programa en MATLAB 6-22

```
num=[1 4 6];
den=[1 5 4];
nyquist(num,den);
grid
title('Diagrama de Nyquist de G(s)=(s^2+4s+6)/(s^2+5s+4)')
```

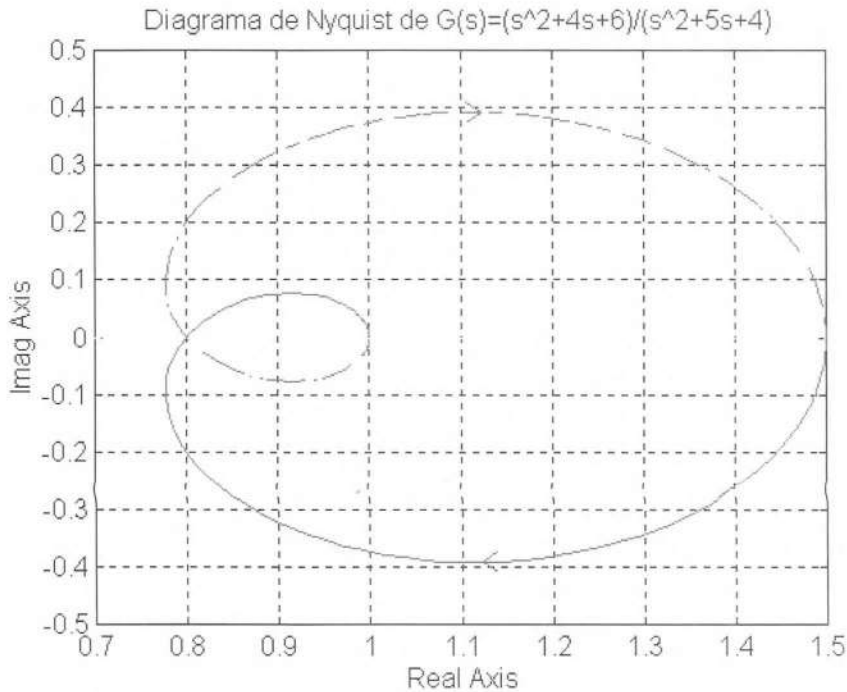


Figura 6-44

EJEMPLO 6-19

Considere el sistema de realimentación positiva con la siguiente función de transferencia en lazo abierto:

$$G(s) = \frac{s^2 + 4s + 6}{s^2 + 5s + 4}$$

Dibuje un diagrama de Nyquist.

El diagrama de Nyquist del sistema de realimentación positiva se puede obtener definiendo num y den como

```
num = [-1  -4  -6]
den = [1  5  4]
```

y utilizando la orden nyquist(num,den). El programa de MATLAB 6-23 da el diagrama de Nyquist que se muestra en la Figura 6-45.

Programa en MATLAB 6-23

```
num=[-1 -4 -6];
den=[1 5 4];
nyquist(num,den);
grid
title('Diagrama de Nyquist de G(s)=-(s^2+4s+6)/(s^2+5s+4)')
```

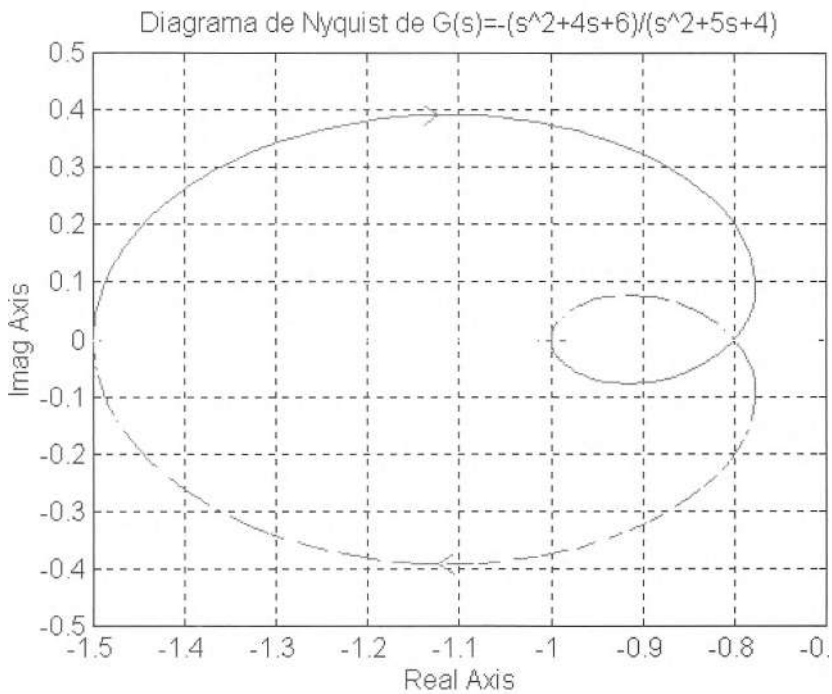


Figura 6-45

El sistema es inestable porque encierra al punto $-1+j0$ una vez en el sentido contrario al de las agujas del reloj. Observe que éste es un caso especial en el que el diagrama de Nyquist pasa por el punto $-1+j0$ y también encierra a este punto una vez en el sentido contrario al de las agujas del reloj. Esto significa que el sistema en lazo cerrado es degenerado; el sistema se comporta como un sistema inestable de primer orden. Véase la siguiente función de transferencia en lazo cerrado del sistema con realimentación positiva:

$$\frac{C(s)}{R(s)} = \frac{s^2 + 4s + 6}{s^2 + 5s + 4 - (s^2 + 4s + 6)} = \frac{s^2 + 4s + 6}{s - 2}$$

Observe que el diagrama de Nyquist para el caso de realimentación positiva es una imagen especular sobre el eje imaginario del diagrama de Nyquist para el caso de realimentación negativa. Esto se puede ver en la Figura 6-46 la cual se obtiene con el programa de MATLAB 6-24.

Programa en MATLAB 6-24

```
num1=[1 4 6];
den1=[1 5 4];
num2=[-1 -4 -6];
den2=[1 5 4];
v=[-2 2 -1 1]; axis(v);
nyquist(num1,den1);
grid
```

```
title('Diagramas de Nyquist de G(s) y de -G(s)')
text(1.05,0.45,'G(s)')
text(0.6,-0.6,'Utilice este Diagrama de')
text(0.6,-0.7,'Nyquist para sistemas con')
text(0.6,-0.8,'realimentación negativa')
hold

%Current plot held

nyquist(num2,den2);
text(-1.3,0.45,'-G(s)')
text(-1.7,-0.6,'Utilice este Diagrama de')
text(-1.7,-0.7,'Nyquist para sistemas con')
text(-1.7,-0.8,'realimentación positiva')
hold

%Current plot released
```

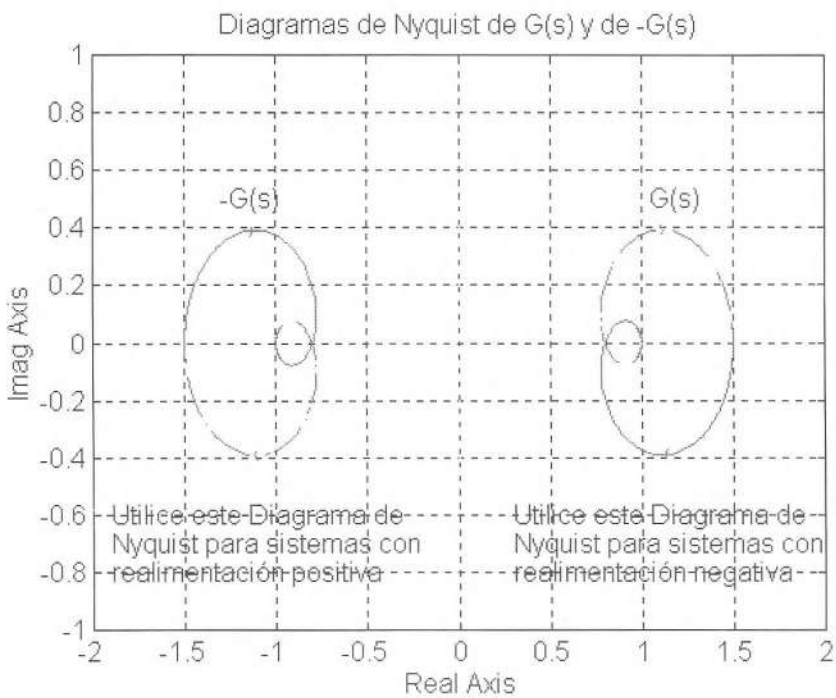


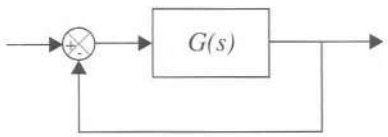
Figura 6-46

Ejemplos adicionales de diagramas de Nyquist

Algunos ejemplos adicionales de diagramas de Nyquist se muestran en los Ejemplos 6-20 al 6-24.

EJEMPLO 6-20

Sistema



$$G(s) = \frac{1}{s(s^2 + s + 0.5)}$$

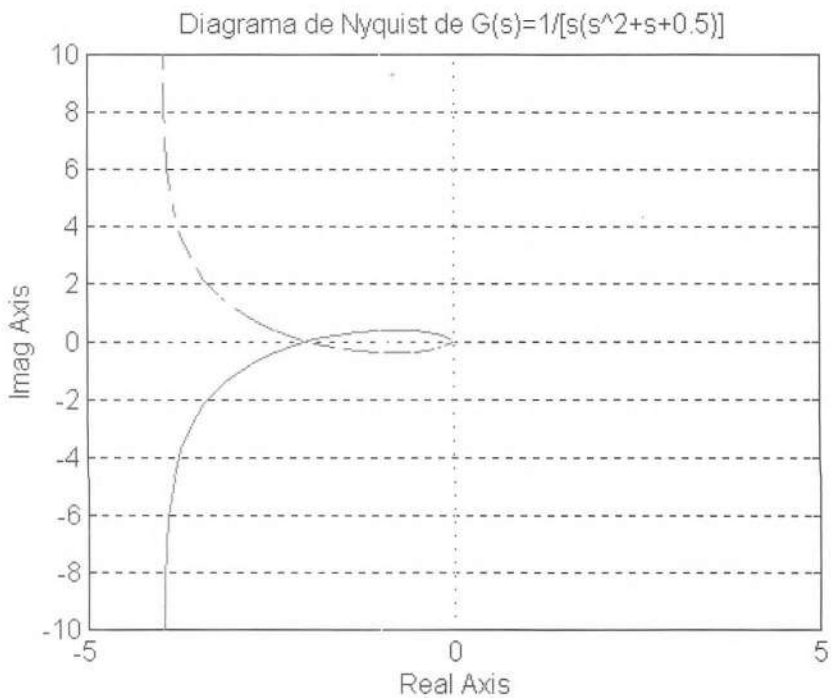
Programa en MATLAB

```
num=[0 0 0 1];
den=[1 1 0.5 0];
v=[-5 5 -10 10];
axis(v);
nyquist(num,den)

Warning: Divide by zero

grid
title('Diagrama de Nyquist de G(s)=1/[s(s^2+s+0.5)]')
```

Estabilidad

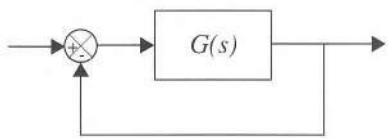
```
El punto -1+j0 se enlaza dos veces en el sentido de las agujas de un.reloj.
El sistema en lazo cerrado es inestable
```

Diagrama de Nyquist



EJEMPLO 6-21

Sistema



$$G(s) = \frac{1}{s(s^2 + 0.8s + 1)}$$

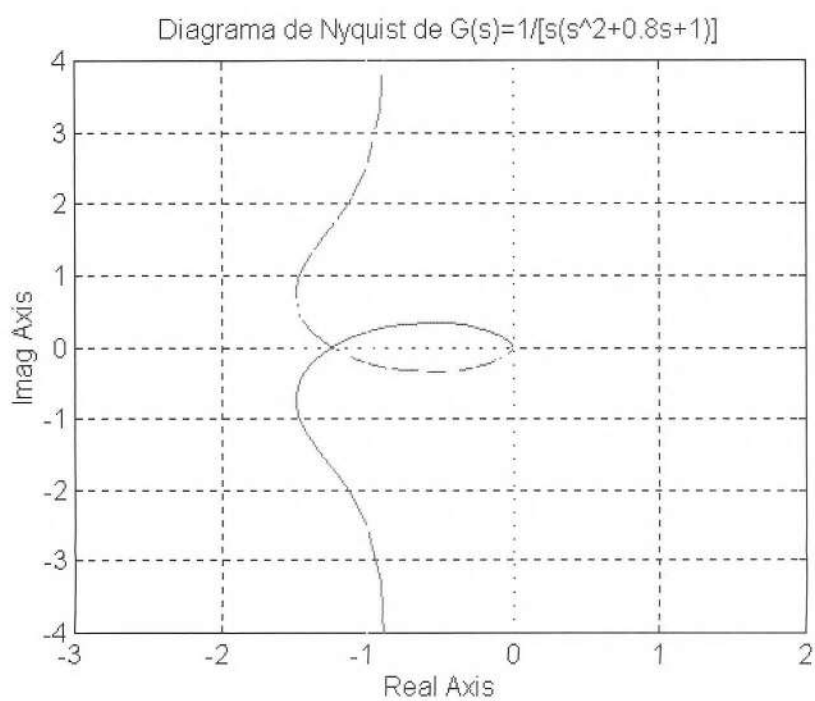
Programa en MATLAB

```
num=[0 0 0 1];
den=[1 0.8 1 0];
v=[-3 2 -4 4];
axis(v);
nyquist(num,den)

Warning: Divide by zero

grid
title('Diagrama de Nyquist de G(s)=1/[s(s^2+0.8s+1)]')
```

Estabilidad

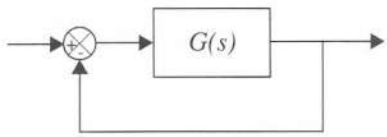
```
El punto -1+j0 se enlaza dos veces en el sentido de las agujas de un.reloj.
El sistema en lazo cerrado es inestable
```

Diagrama de Nyquist



EJEMPLO 6-22

Sistema



$$G(s) = \frac{20(s+1)}{s(s+5)(s^2+2s+10)}$$

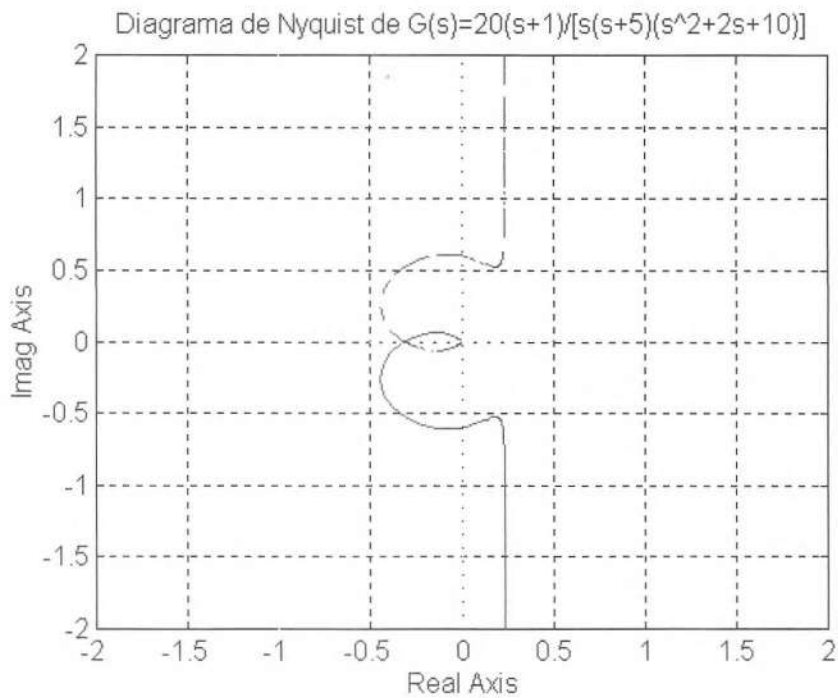
Programa en MATLAB

```
num=[0 0 0 20 20];
den=[1 7 20 50 0];
v=[-2 2 -2 2];
axis(v);
nyquist(num,den)

Warning: Divide by zero

grid
title('Diagrama de Nyquist de G(s)=20(s+1)/[s(s+5)(s^2+2s+10)]')
```

Estabilidad

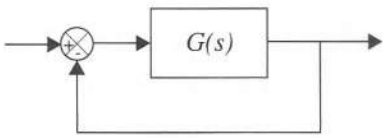
```
El punto -1+j0 no se enlaza ninguna vez.
El sistema en lazo cerrado es estable
```

Diagrama de Nyquist



EJEMPLO 6-23

Sistema



$$G(s) = \frac{s + 0.5}{(s^2 + s + 1)}$$

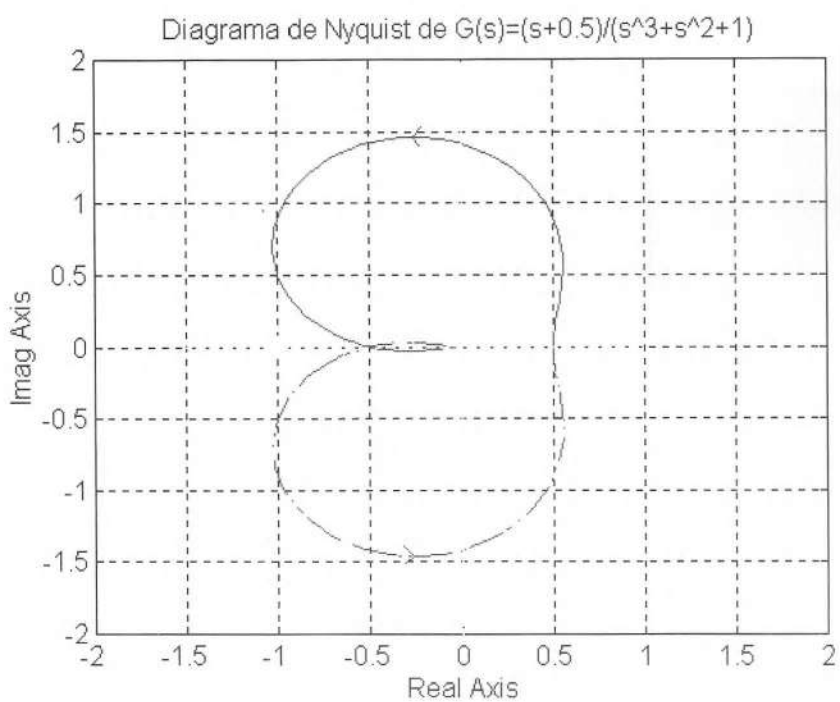
Programa en MATLAB

```
num=[0 0 1 0.5];
den=[1 1 0 1];
v=[-2 2 -2 2];
axis(v);
nyquist(num,den)
grid
title('Diagrama de Nyquist de G(s)=(s+0.5)/(s^3+s^2+1)')
```

Estabilidad

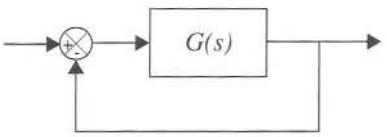
```
Los polos en lazo abierto estan en el semiplano derecho s.
El punto -1+j0 no se enlaza ninguna vez.
El sistema en lazo cerrado es inestable
```

Diagrama de Nyquist



EJEMPLO 6-24

Sistema



$$G(s) = \frac{100(s+0.5)}{s^2(s+2)(s+10)}$$

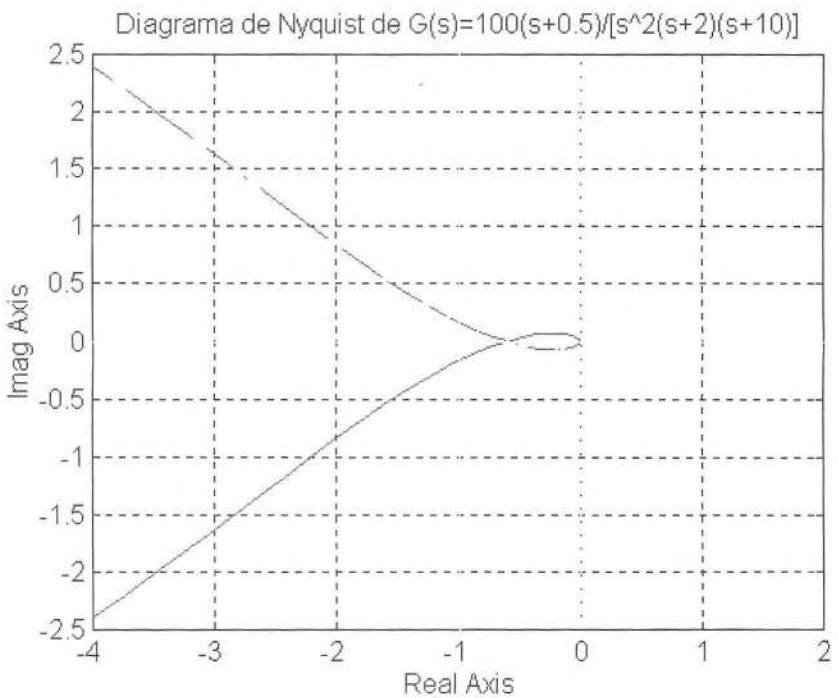
Programa en MATLAB

```
num=[0 0 0 100 50];
den=[1 12 20 0 0];
v=[-4 2 -2.5 2.5];
axis(v);
nyquist(num,den)

Warning: Divide by zero

grid
title('Diagrama de Nyquist de G(s)=100(s+0.5)/[s^2(s+2)(s+10)]')
```

Estabilidad

```
El punto -1+j0 no se enlaza ninguna vez.
El sistema en lazo cerrado es estable
```

Diagrama de Nyquist



Dibujo del diagrama de Nyquist de sistemas definidos en el espacio de estados

Considere el sistema definido por

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

donde

$\mathbf{x}$ = vector de estado (n-vector)
$\mathbf{y}$ = vector salida (m-vector)
$\mathbf{u}$ = vector de control (r-vector)
$\mathbf{A}$ = matriz de estado (matriz de $n \times n$)
$\mathbf{B}$ = matriz de control (matriz de $n \times r$)
$\mathbf{C}$ = matriz de salida (matriz de $m \times n$)
$\mathbf{D}$ = matriz de transmisión directa ($m \times r$)

Los diagramas de Nyquist para este sistema se pueden obtener introduciendo la orden

```
nyquist(A,B,C,D)
```

Esta orden produce una serie de diagramas de Nyquist, uno por cada combinación de entrada-salida del sistema. El rango de frecuencias se determina de forma automática.

La orden

```
nyquist(A,B,C,D,iu)
```

produce diagramas de Nyquist de una única entrada iu a todas las salidas del sistema, con el rango de frecuencias determinado de forma automática. El escalar iu es un índice dentro de las entradas del sistema y especifica qué entrada utilizar para la respuesta en frecuencia.

La orden

```
nyquist(A,B,C,D,iu,w)
```

utiliza un vector w de frecuencias definido por el usuario. El vector w especifica las frecuencias en radianes por segundo en las cuales la respuesta en frecuencia será calculada.

EJEMPLO 6-25

Considere el sistema definido por

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} u$$

Dibuje un diagrama de Nyquist.

Este sistema tiene una única entrada u y una única salida y. Un diagrama de Nyquist se puede obtener introduciendo la orden

```
nyquist(A,B,C.D)
```

o

```
nyquist(A,B,C,D,1)
```

El programa de MATLAB 6-25 dará el diagrama de Nyquist. (Observe que se obtiene un resultado idéntico con cualquiera de las dos órdenes de arriba). La Figura 6-47 muestra el diagrama de Nyquist del programa de MATLAB 6-25.

Si desea dibujar el diagrama de Nyquist en un dominio mayor, por ejemplo [-2 2 -2 2], puede introducir el programa de MATLAB 6-26 en el computador. El gráfico resultante se muestra en la Figura 6-48.

Programa en MATLAB 6-25

```
A=[0 1;-25 -4];
B=[0;25];
C=[1 0];
D=[0];
nyquist(A,B,C,D)
grid
title('Diagrama de Nyquist')
```

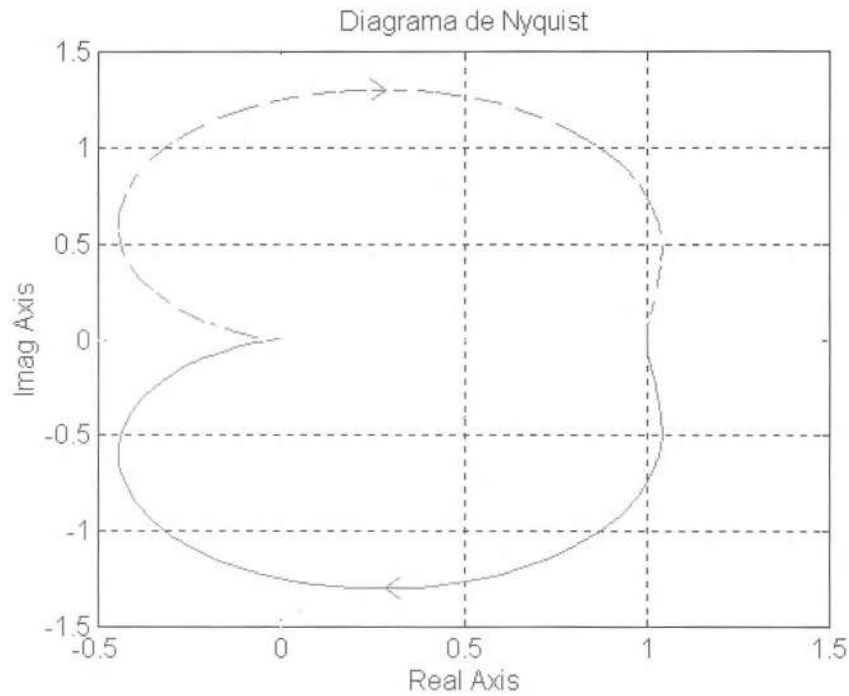


Figura 6-47

Programa en MATLAB 6-26

```
A=[0 1;-25 -4];
B=[0;25];
C=[1 0];
D=[0];
v=[-2 2 -2 2]; axis(v);
nyquist(A,B,C,D)
grid, title('Diagrama de Nyquist')
```

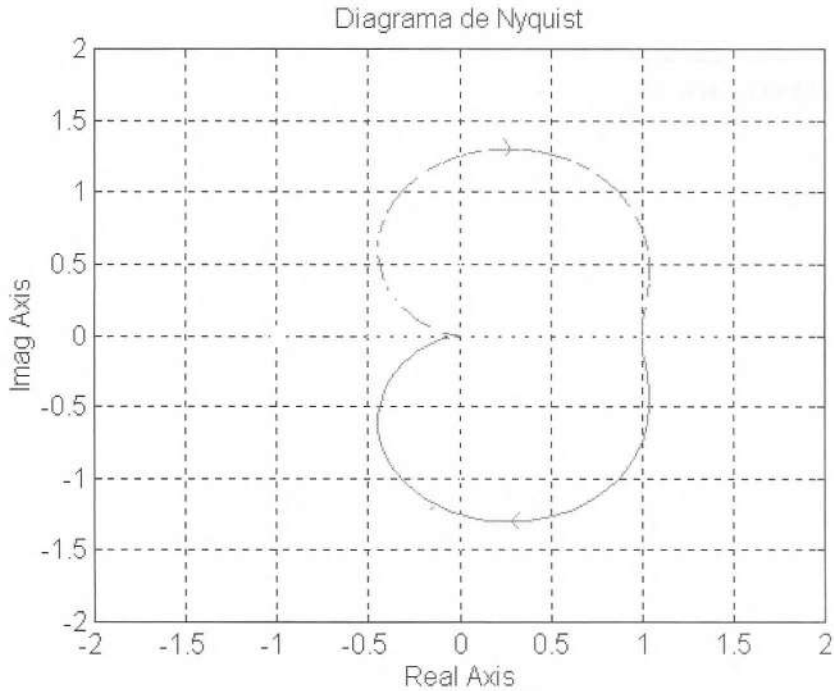


Figura 6-48

EJEMPLO 6-26

Considere el sistema definido por

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -1 & -1 \\ 6{,}5 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

Este sistema tiene dos entradas y dos salidas. Existen cuatro relaciones sinusoidales salida-entrada: $Y_1(j\omega)/U_1(j\omega)$, $Y_2(j\omega)/U_1(j\omega)$, $Y_1(j\omega)/U_2(j\omega)$ e $Y_2(j\omega)/U_2(j\omega)$. Dibuje los diagramas de Nyquist para el sistema. (Cuando se considera la entrada u_1 se supone que u_2 es cero y viceversa).

Los cuatro gráficos individuales de Nyquist se pueden obtener utilizando la orden

nyquist(A,B,C,D)

El programa de MATLAB 6-27 produce los cuatro diagramas de Nyquist. Se muestran en la Figura 6-49. [Dependiendo de la versión de MATLAB (versión Estudiante o versión Profesional 4.0) se pueden obtener gráficos ligeramente diferentes].

Para dibujar dos diagramas de Nyquist para la entrada u_1 en un único gráfico y dos diagramas de Nyquist para la entrada u_2 en otro gráfico, utilice las órdenes

nyquist(A,B,C,D,1) y nyquist(A,B,C,D,2)

respectivamente

Programa en MATLAB 6-27

```
A=[-1 -1;6.5 0];
B=[1 1;1 0];
C=[1 0;0 1];
D=[0 0;0 0];
v=[-2 2 -2 2];
nyquist(A,B,C,D)
```

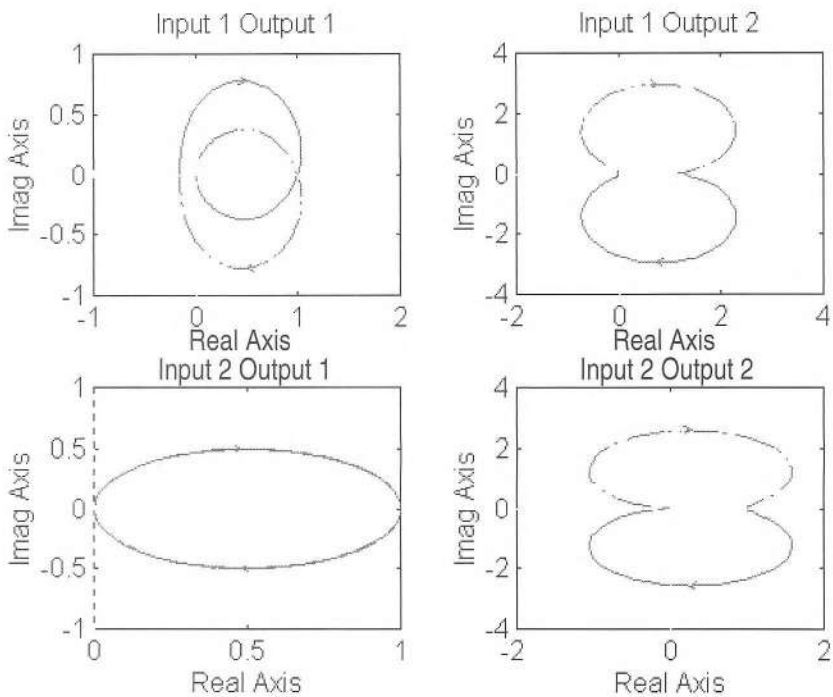


Figura 6-49

El programa de MATLAB 6-28 dibujará dos diagramas de Nyquist para la entrada u_1 en un único gráfico y dos diagramas de Nyquist para la entrada u_2 en otro gráfico. La Figura 6-50 muestra los dos gráficos.

La orden

$$[re,im,w] = nyquist(A,B,C,D,iu)$$

devuelve la respuesta en frecuencia en las matrices re, im y w. Esta orden no dibuja en pantalla. Las matrices re e im contienen las partes reales e imaginarias de la respuesta en frecuencia del sistema evaluados en los puntos de frecuencia de w. Observe que re e im tienen tantas columnas como salidas y una fila por cada elemento de w.

Si se desea dibujar el diagrama de Nyquist para $Y_1(jw)/U_1(jw)$ únicamente, utilice esta orden.

Es decir, introduzca la orden

$$[re,im,w] = nyquist(A,B,C,D,1)$$

En el sistema considerado en el ejemplo, re es una matriz de 2 columnas.

Programa en MATLAB 6-28

```
%-------Diagramas de Nyquist-------

%*****En este programa se obtienen los diagramas de Nyquist de un sistema con
%dos entradas
%(u1 y u2) y dos salidas (y1 e y2)*****

%*****Primero, se obtendrán los diagramas de Nyquist cuando la entrada es u1.
%Después se obtendrán los diagramas de Nyquist cuando la entrada es u2*****

%*****Introduzca las matrices A,B,C y D*****

A=[-1 -1;6.5 0];
B=[1 1;0 1];
C=[1 0;0 1];
D=[0 0;0 0];

%*****Para obtener los diagramas de Nyquist cuando la entrada es u1, introduzca
%la orden 'nyquist(A,B,C,D,1)'*****

nyquist(A,B,C,D,1)
grid
title('Diagramas de Nyquist: Entrada=u1 (u2=0)')
text(0.1,0.7,'Y1')
text(0.1,2.5,'Y2')

%*****A continuación, se obtendrán los diagramas de Nyquist cuando la entrada
%es u2. Introduzca la orden 'nyquist(A,B,C,D,2)'*****

nyquist(A,B,C,D,2)
grid
title('Diagramas de Nyquist: Entrada=u2 (u1=0)')
text(0.1,0.5,'Y1')
text(0.1,2.2,'Y2')
```

La primera columna está formada por las partes reales de Y_1 cuando los puntos de frecuencia se mueven y la segunda columna por las partes reales de Y_2. Para obtener la primera columna (las partes reales de Y_1, definidas como re1), introduzca la declaración:

```
re1 = re*[1;0]
```

Análogamente, para obtener la parte imaginaria de Y_1, introduzca la declaración

```
im1 = im*[1;0]
```

Para obtener el diagrama de Nyquist para $Y_1(jw)/U_1(jw)$ introduzca la siguiente orden *plot*:

```
plot(re1,im1)
```

El programa de MATLAB 6-29 obtiene el diagrama de Nyquist para $Y_1(jw)/U_1(jw)$ para $w > 0$ que se muestra en la Figura 6-51. Para dibujar el lugar de Nyquist para $-\infty < w < \infty$, sustituya la orden plot(re1,im1) por plot(re1,im1,re1,-im1).

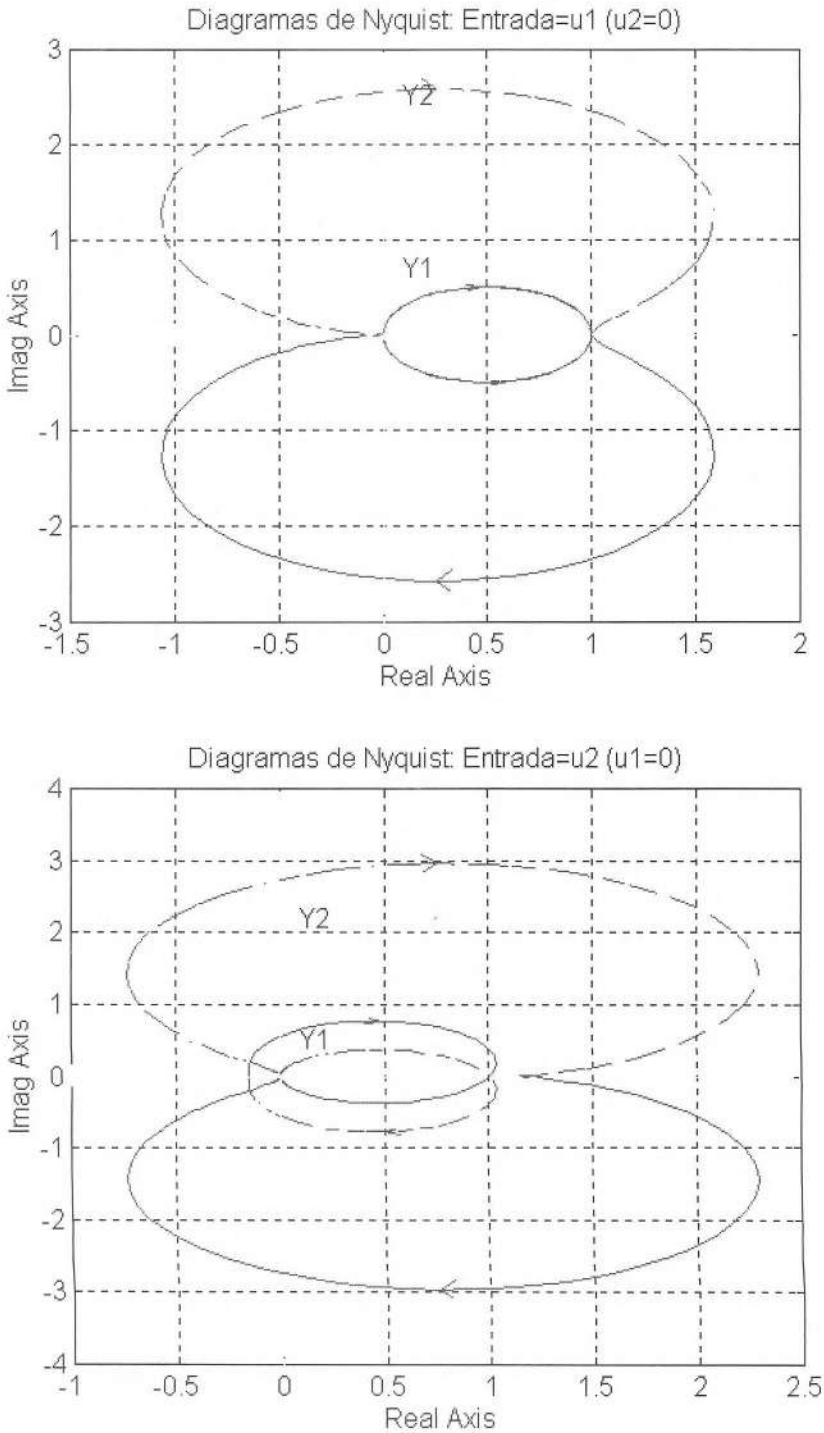

Diagramas de Nyquist: Entrada=u1 (u2=0)
Y2
Y1
Imag Axis
Real Axis
3
2
1
0
-1
-2
-3
-1.5
-1
-0.5
0
0.5
1
1.5
2
Diagramas de Nyquist: Entrada=u2 (u1=0)
Y2
Y1
Imag Axis
Real Axis
4
3
2
1
0
-1
-2
-3
-4
-1
-0.5
0
0.5
1
1.5
2
2.5

Figura 6-50

Programa en MATLAB 6-29

```
A=[-1 -1;6.5 0];
B=[1 1;1 0];
C=[1 0;0 1];
D=[0 0;0 0];
[re,im,w]=nyquist(A,B,C,D,1);
re1=re*[1;0];
im1=im*[1;0];
plot(re1,im1)
grid
title('Diagrama de Nyquist para Y1(jw)/U1(jw)')
xlabel('Eje real')
ylabel('Eje imaginario')
```

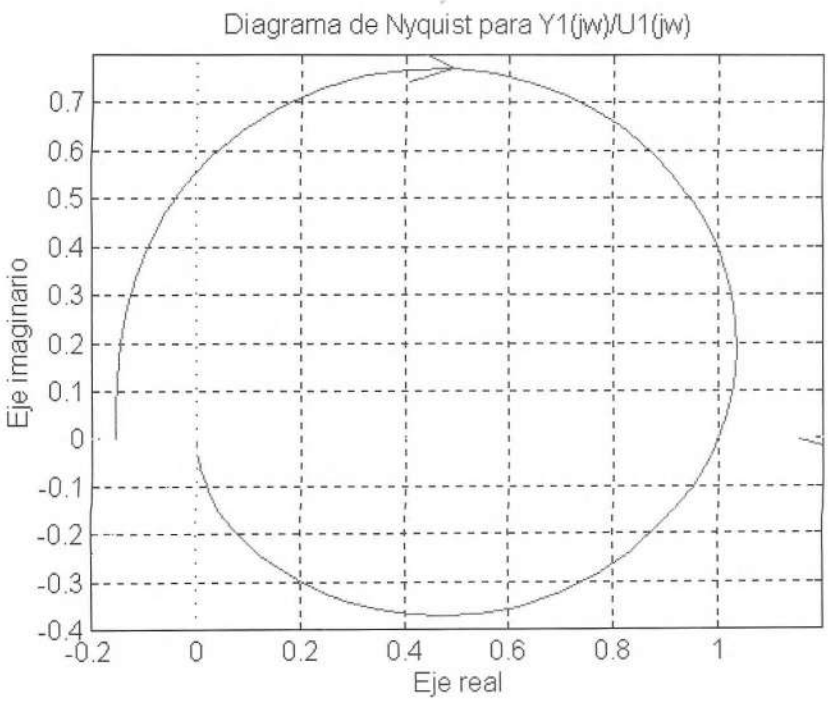


Figura 6-51

6.5 EJEMPLOS DE APLICACIÓN

En esta sección se discute el diseño de sistemas en lazo cerrado basado en la aproximación del diagrama de Bode. Esta aproximación para el diseño de sistemas de control es particularmente útil por las siguientes razones:

1. En el diagrma de Bode, la asíntota de baja frecuencia de la curva de magnitud indica una de las constantes de error estáticas K_p, K_v o K_a.

2. *Las especificaciones de la respuesta transitoria se pueden traducir a las de respuesta en frecuencia en* términos de margen de fase, margen de ganancia, ancho de banda y así sucesivamente. Estas especificaciones

se pueden manejar fácilmente en el diagrama de Bode. En particular, los márgenes de fase y de ganancia se pueden leer directamente del diagrama de Bode.

3. El diseño de un compensador o controlador para satisfacer las especificaciones dadas (en términos de margen de fase y margen de ganancia) se pueden llevar a cabo en el diagrama de Bode de una manera simple y directa.

Antes de presentar los problemas de diseño, se van a revisar las técnicas de compensación de adelanto de fase, retraso de fase y retardo-adelanto de fase.

La compensación por adelanto de fase se usa comúnmente para mejorar los márgenes de estabilidad. La compensación por adelanto de fase incrementa el ancho de banda del sistema. Asi, el sistema tendrá una velocidad de respuesta más rápida. Sin embargo, cuando un sistema utiliza compensación por adelanto de fase, está sujeto a problemas de ruidos en alta frecuencia debido a su incremento en las ganancias de alta frecuencia.

La compensación por retardo de fase reduce la ganancia del sistema a altas frecuencias sin reducir la ganancia del sistema a bajas frecuencias. Se reduce el ancho de banda del sistema y el sistema tiene una velocidad de respuesta más lenta. Al reducirse la ganancia en alta frecuencia, se puede incrementar la ganancia total del sistema y por tanto se puede incrementar la ganancia a bajas frecuencias y mejorar la precisión en el estado estacionario. También es posible atenuar cualquier ruido a alta frecuencia que tenga el sistema.

En algunas aplicaciones, un compensador de retardo de fase se encuentra en cascada con un compensador de adelanto de fase. El compensador en cascada se conoce como compensador *retardo-adelanto de fase*. Al utilizar el compensador retardo-adelanto de fase se puede incrementar la ganancia a bajas frecuencias (lo que significa una mejora en la precisión del estado estacionario), mientras al mismo tiempo se incrementa el ancho de banda y los márgenes de estabilidad del sistema.

Observe que el controlador PID es un caso especial de un controlador retardo-adelanto de fase. La acción de control PD, la cual afecta a la región de altas frecuencias, incrementa el ángulo del adelanto de fase y mejora la estabilidad del sistema, también incrementa el ancho de banda del sistema (incrementando la velocidad de respuesta). Es decir, el controlador PD tiene un comportamiento de compensador por adelanto de fase. La acción de control PI afecta a la parte de bajas frecuencias, de hecho incrementa la ganancia a bajas frecuencias y mejora la precisión del estado estacionario. Por tanto, el controlador PI actúa como un compensador de retardo de fase. La acción de control PID es una combinación de las acciones de control PI y PD. Las técnicas de diseño para controladores PID son básicamente la de los compensadores retardo-adelanto de fase. (Sin embargo, en sistemas de control industrial cada una de las acciones en el controlador PID se pueden ajustar experimentalmente).

A continuación se discute primero un sencillo problema de ajuste de ganancia en el Ejemplo 6-27. El Ejemplo 6-28 trata el diseño de un compensador de adelanto de fase. El Ejemplo 6-29 trata el diseño de un compensador de retardo-adelanto de fase. Aquí se utilizarán los diagramas de Bode y Nyquist para verificar el diseño basado en métodos convencionales. El alcance de esta sección se limita al uso de MATLAB para comprobar los resultados de los sistemas diseñados. El diseño automático de MATLAB de compensadores no se discute aquí.

EJEMPLO 6-27

Considere el sistema de control mostrado en la Figura 6-52. Determine el valor de la ganancia K para que el margen de ganancia sea 60°.

La función de transferencia en lazo abierto es

$$G(s) = K\,\frac{s+0,1}{s+0,5}\,\frac{10}{s(s+1)}$$

$$= \frac{K(10s+1)}{s^3+1,5s^2+0,5s}$$

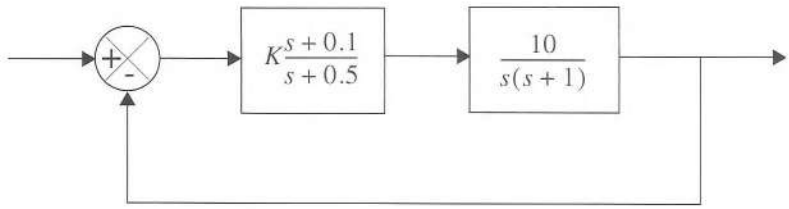


Figura 6-52

Vamos a dibujar el diagrama de Bode de $G(s)$ cuando $K = 1$. Se puede utilizar el programa de MATLAB 6-30 para este propósito. La Figura 6-53 muestra el diagrama de Bode que da este programa. De este diagrama, el margen de fase requerido de 60° ocurre en la frecuencia w = 1.15 rad/seg. La magnitud de $G(j\omega)$ a esta frecuencia es de 14.5 dB. Por tanto la ganancia debe de satisfacer la siguiente ecuación:

$$20 \log K = -14.5 \text{ dB}$$

o

$$K = 0.188$$

Programa en MATLAB 6-30

```
num=[0 0 10 1];
den=[1 1.5 0.5 0];
bode(num,den)
title('Diagrama de Bode de G(s)=(10s+1)/[s(s+0.5)(s+1)]')
```

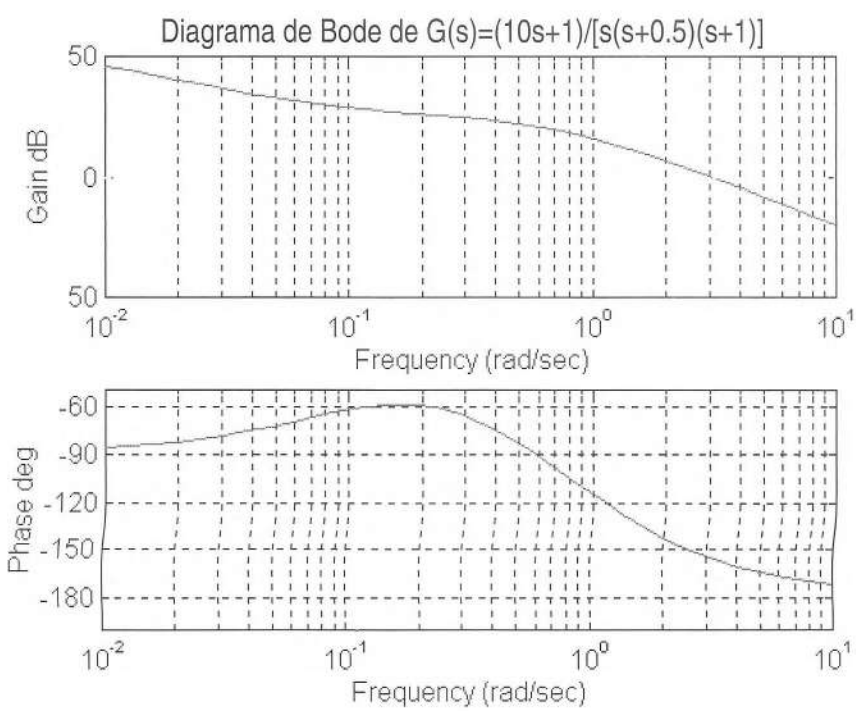


Figura 6-53

Por tanto se ha determinado el valor de la ganancia K.

Para verificar los resultados, se va a dibujar el diagrama de Nyquist de G para el rango de frecuencias

$$w = 0.1:0.01:1.15$$

El punto final del lugar (w = 1.15 rad/seg) estará sobre el círculo unidad en el plano de Nyquist.

Para comprobar el margen de fase, es conveniente dibujar el diagrama de Nyquist sobre un diagrama polar utilizando las rejillas polares.

Para dibujar el diagrama de Nyquist sobre un diagrama polar, defina el vector complejo z como

$$z = re + i*im = re^{i\theta}$$

donde r y θ (teta) vienen dados por

$$r = abs(z)$$
$$teta = angle(z)$$

abs significa la raíz cuadrada de la suma de la parte real al cuadrado y de la parte imaginaria al cuadrado; angle significa $\tan^{-1}$(parte imaginaria/parte real).

Si se utiliza la orden

$$polar(teta,r)$$

Programa en MATLAB 6-31

```
%-------Diagrama de Nyquist en coordenadas polares-------

num=[0 0 1.88 0.188];
den=[1 1.5 0.5 0];
w=0.1:0.05:1.15;
v=[-2 2 -2 2];
axis(v);axis('square');
[re,im,w]=nyquist(num,den,w);

%*****Convierta las coordenadas rectangulares en coordenadas polares definiendo
%z, r y teta de la manera siguiente*****

z=re+i*im;
r=abs(z);
teta=angle(z);

%*****Para dibujar el diagrama polar, introduzca la orden 'polar(teta,r)'*****

polar(teta,r)
grid
title('Comprueba el margen de fase');
text(-1.65,1.8,'Diagrama de Nyquist')
text(-3,1.5,'El margen de fase')
text(-2.9,1.6,'es de 60 grados')

%*****Restaure la perspectiva gráfica normal*****

axis('normal')
```

MATLAB producirá un diagrama en coordenadas polares. A continuación, la orden *grid* dibujará las líneas y círculos de la rejilla polar.

El programa de MATLAB 6-31 produce el diagrama de Nyquist de $G(jw)$, donde w se encuentra entre 0.1 rad/seg y 1.15 rad/seg. El diagrama resultante se muestra en la Figura 6-54. Observe que el punto $G(j1.15)$ se encuentra sobre el círculo unidad y el ángulo de fase de este punto es 120°. Por tanto el margen de fase es 60°. El hecho de que el punto $G(j1.15)$ esté sobre el círculo unidad verifica que en w = 1.15 rad/seg la magnitud es igual a 1 o a 0 dB. (Por tanto w = 1.15 es la frecuencia de la ganancia de cruce). Así $K = 0.188$ nos da el margen de fase deseado de 60°.

Observe que para escribir texto en un diagrama polar, se introduce la orden *text* como sigue:

text(ángulo en radianes,radio,' ')

Por ejemplo, para escribir 'Diagrama de Nyquist'comenzando en el punto (angle = -1.65 rad, radio = 1.8), introduzca la orden:

text(-1.65,1.8,'Diagrama de Nyquist')

El texto se escribe horizontalmente en la pantalla.

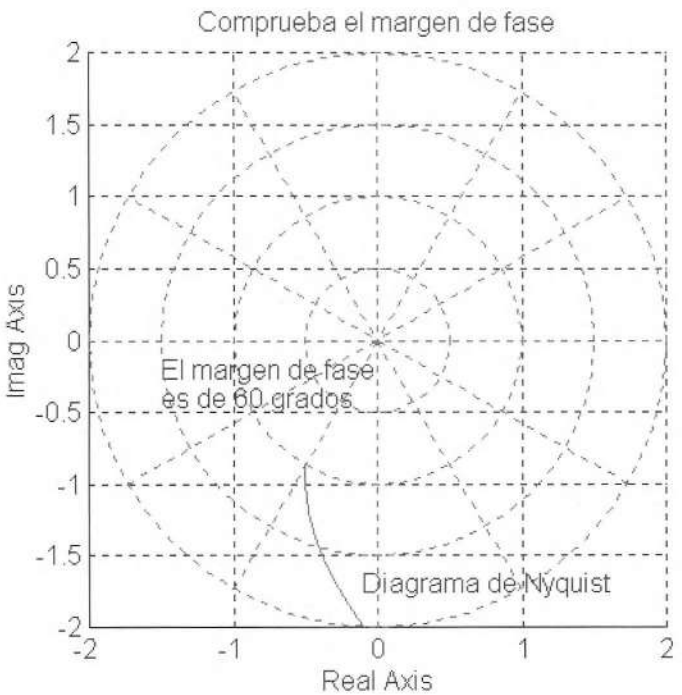
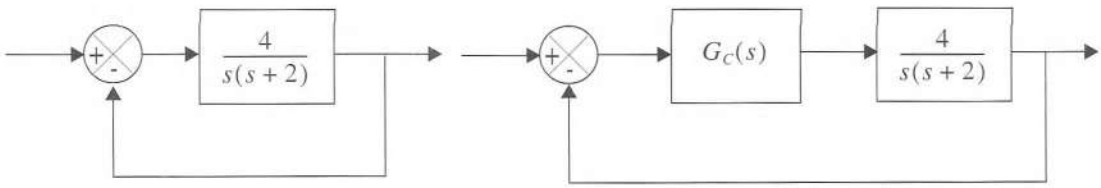


Figura 6-54

EJEMPLO 6-28: DISEÑO DEL COMPENSADOR DE ADELANTO

Considere el sistema mostrado en la Figura 6-55(a). Se desea diseñar un compensador tal que el sistema en lazo cerrado satisfaga las siguientes condiciones:

constante de error de velocidad estática = $K_v = 20\ \text{seg}^{-1}$
margen de fase = 50°
margen de ganancia ≥ 10 dB



(a) (b)

Figura 6-55

Se diseñará un compensador de adelanto de fase $G_c(s)$ de la forma

$$G_c(s) = K_c\alpha \frac{Ts+1}{\alpha Ts+1} = K_c \frac{s+\frac{1}{T}}{s+\frac{1}{\alpha T}}$$

El sistema compensado se muestra en la Figura 6-55(b)
Defina

$$G_1(s) = KG(s) = \frac{4K}{s(s+2)}$$

donde $K = K_c\alpha$.

El primer paso en el diseño es ajustar la ganancia K para encontrar la especificación de comportamiento de estado estacionario o proporcionar la constante de error de velociadad estática requerida. Como la constante de error K_v viene dada como 20 seg^{-1}, se tiene

$$K_v = \lim_{s\to 0} sG_c(s)G(s) = \lim_{s\to 0} s\frac{Ts+1}{\alpha Ts+1}G_1(s)$$

$$= \lim_{s\to 0} \frac{s4K}{s(s+2)} = 2K = 20$$

o

$$K = 10$$

Con $K = 10$, el sistema compensado cumplirá el requisito de estado estacionario.
A continuación representaremos el diagrama de Bode de

$$G_1(s) = \frac{40}{s(s+2)}$$

El programa de MATLAB 6-32 produce el diagrama de Bode que se muestra en la Figura 6-56. En este diagrama se encuentra que el margen de fase es de 17°. El margen de ganancia es de $+\infty$ dB.

Como la especificación busca un margen de fase de 50°, el adelanto de fase adicional necesario para satisfacer el requisito de margen de fase es 33°. Un compensador de adelanto puede contribuir a esta cantidad.

Teniendo en cuenta que la adición de un compensador de adelanto modifica la curva de magnitud en el diagrama de Bode, observamos que la frecuencia de la ganancia de cruce se desplazará hacia la derecha. Debemos compensar el aumento en el retardo de fase de $G_1(jw)$ debido a este aumento en la frecuencia de la ganancia de cruce. Tomando en consideración el desplazamiento en la frecuencia de la ganancia de cruce, podemos suponer que ϕ_m, el máximo adelanto de fase requerido, es aproximadamente de 38°. (Esto significa que se han añadido unos 5° para compensar el desplazamiento en la ganancia de la frecuencia de cruce.). Como $\phi_m = 38°$ corresponde a $\alpha = 0.2379$. Observe que $\alpha = 0.24$ corresponde a $\phi_m = 37.8°$. Si seleccionamos $\phi_m = 38°$ o $\phi_m = 37.8°$ no hay mucha diferencia en la solución final. Por tanto seleccionamos $\alpha = 0.24$.

$$\operatorname{sen}\phi_m = \frac{1-\alpha}{1+\alpha}$$

Programa en MATLAB 6-32

```
num=[0 0 40];
den=[1 2 0];
w=logspace(-1,1,100);
bode(num,den,w)
title('Diagrama de Bode de G1(s)=40/[s(s+2)]')
```

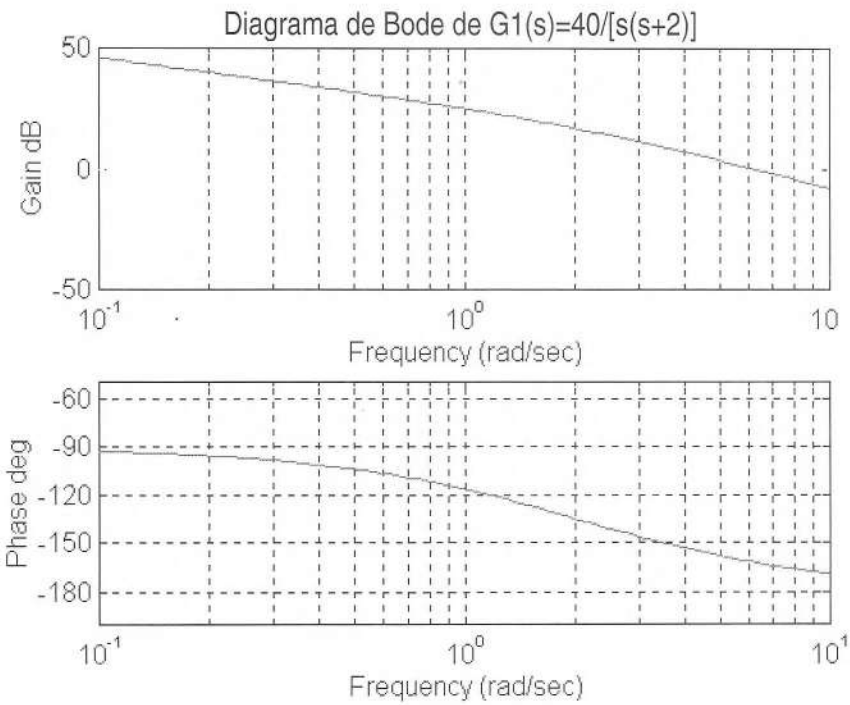


Figura 6-56

Una vez que ha sido determinada la atenuación α a partir del ángulo de fase de adelanto requerido, el siguiente paso es determinar las frecuencias esquinas $w = 1/T$ y $w = 1/(\alpha T)$ del compensador de adelanto. Obsérvese que el máximo ángulo de adelanto de fase ϕ_m ocurre en la media geométrica de las dos frecuencias esquinas, o $w = 1/(\sqrt{\alpha}T)$.

La cantidad de la modificación en la curva de magnitud en $w = 1/(\sqrt{\alpha}T)$ debido a la inclusión del término $(Ts + 1)/(\alpha Ts + 1)$ es

$$\left|\frac{1+j\omega T}{1+j\omega\alpha T}\right|_{\omega=\frac{1}{T\sqrt{\alpha}}} = \left|\frac{1+j\dfrac{1}{\sqrt{\alpha}}}{1+j\alpha\dfrac{1}{\sqrt{\alpha}}}\right| = \frac{1}{\sqrt{\alpha}}$$

Observe que

$$\frac{1}{\sqrt{\alpha}} = \frac{1}{\sqrt{0{,}24}} = 0{,}2041 = 6{,}2 \text{ dB}$$

Necesitamos encontrar el punto de frecuencia donde, al añadir el compensador de adelanto, la magnitud total es de 0 dB.

De la Figura 6-56 vemos que el punto de frecuencia donde la magnitud de $G_1(jw)$ es -6.2 dB ocurre entre w =1 y 10 rad/seg. Por tanto representamos un nuevo diagrama de Bode de $G_1(jw)$ en el rango de frecuencias entre w = 1 y 10 para localizar el punto exacto donde $G_1(jw)$ = -6.2 dB. El programa de MATLAB 6-33 produce el diagrama de Bode en este rango de frecuencias que se muestra en la Figura 6-57. De este diagrama, encontramos que el punto de frecuencia donde $|G_1(jw)|$ = -6.2 dB ocurre en w = 9 rad/seg. Seleccionamos esta frecuencia para que sea la nueva frecuencia de la ganancia de cruce, o w_c = 9 rad/seg. Observando que esta frecuencia corresponde a $1/(T\sqrt{\alpha})$, o

$$\omega_c = \frac{1}{T\sqrt{\alpha}}$$

obtenemos

$$\frac{1}{T} = \omega_c\sqrt{\alpha} = 9\sqrt{0{,}24} = 4{,}409$$

y

$$\frac{1}{\alpha T} = \frac{\omega_c}{\sqrt{\alpha}} = \frac{9}{\sqrt{0{,}24}} = 18{,}371$$

Programa en MATLAB 6-33

```
num=[0 0 40];
den=[1 2 0];
w=logspace(0,1,100);
bode(num,den,w)
title('Diagrama de Bode de G1(s)=40/[s(s+2)]')
```

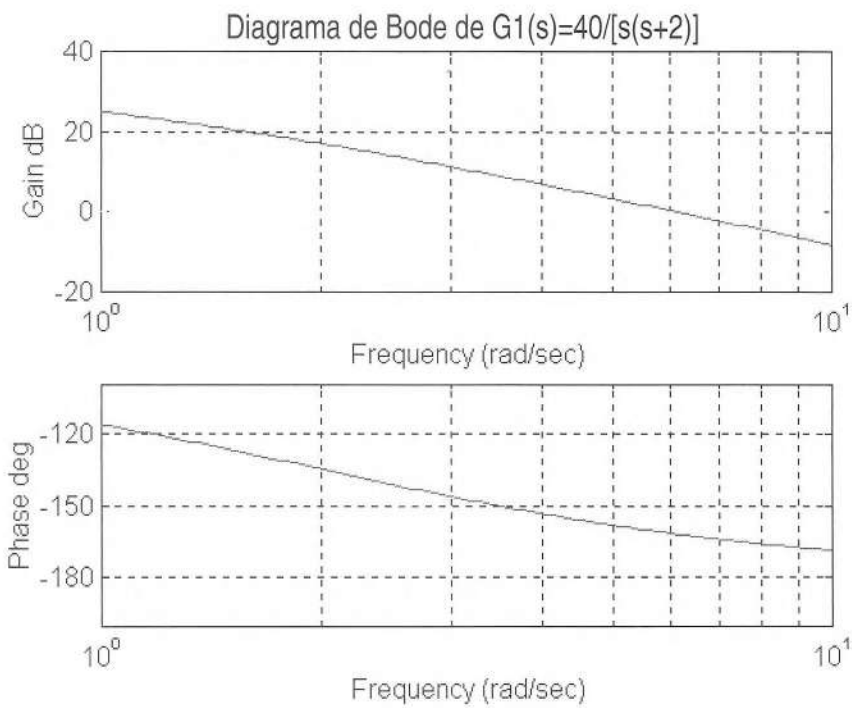


Figura 6-57

El compensador de adelanto así determinado es

$$G_c(s) = K_c \frac{s + 4{,}409}{s + 18{,}371} = K_c\alpha \frac{0{,}227s + 1}{0{,}0544s + 1}$$

donde K_c se determina como

$$K_c = \frac{K}{\alpha} = \frac{10}{0{,}24} = 41{,}667$$

Así, la función de transferencia del compensador se hace

$$G_c(s) = 41{,}667 \frac{s + 4{,}409}{s + 18{,}371} = 10 \frac{0{,}227s + 1}{0{,}0544s + 1}$$

El programa de MATLAB 6-34 produce el diagrama de Bode de este compensador de adelanto, que se muestra en la Figura 6-58. Observe que

$$\frac{G_c(s)}{K}G_1(s) = \frac{G_c(s)}{10}G(s) = G_c(s)G(s)$$

Programa en MATLAB 6-34

```
numc=[41.667 183.71];
denc=[1 18.371];
w=logspace(-1,3,100);
bode(numc,denc,w)
title('Diagrama de Bode de Gc(s)=41.667(s+4.409)/(s+18.371)')
```

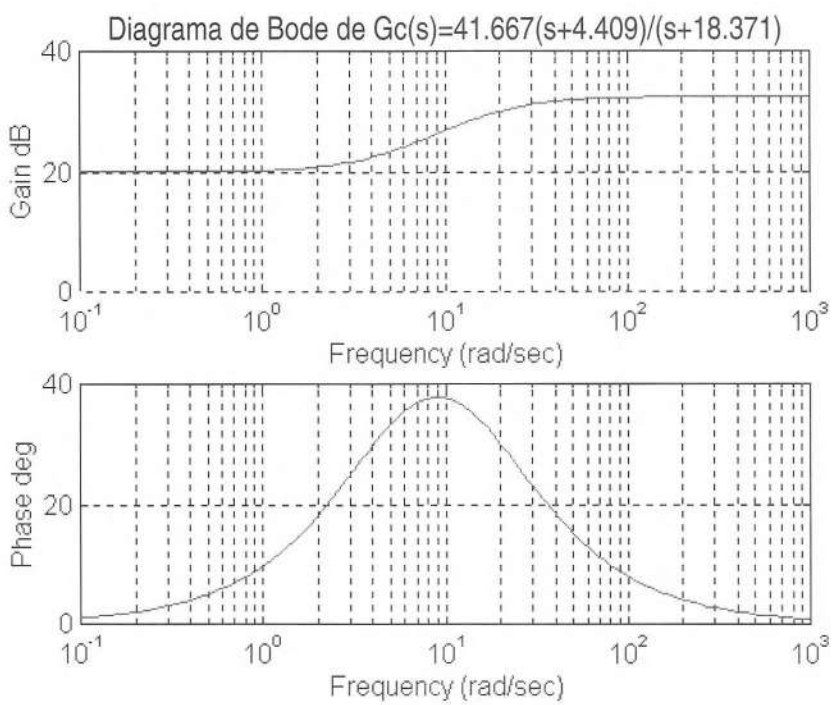


Figura 6-58

La función de transferencia en lazo abierto del sistema diseñado es

$$G_c(s)G(s) = 41{,}667\ \frac{s+4{,}409}{s+18{,}371}\ \frac{4}{s(s+2)}$$

$$= \frac{166{,}668s + 734{,}839}{s^3 + 20{,}371s^2 + 36{,}742s}$$

El programa de MATLAB 6-35 producirá el diagrama de Bode de $G_c(s)G(s)$ que se muestra en la Figura 6-59.

También el programa de MATLAB 6-36 se puede utilizar para dibujar los diagramas de Nyquist de $G(j\omega)$, $G_1(j\omega)$ y $G_c(j\omega)G(j\omega)$ sobre un plano polar. En la Figura 6-60 se muestran los diagramas resultantes.

Programa en MATLAB 6-35

```
num=[0 0 166.668 734.839];
den=[1 20.371 36.742 0];
w=logspace(-1,3,100);
bode(num,den,w)
title('Diagrama de Bode de Gc(s)G(s)')
```

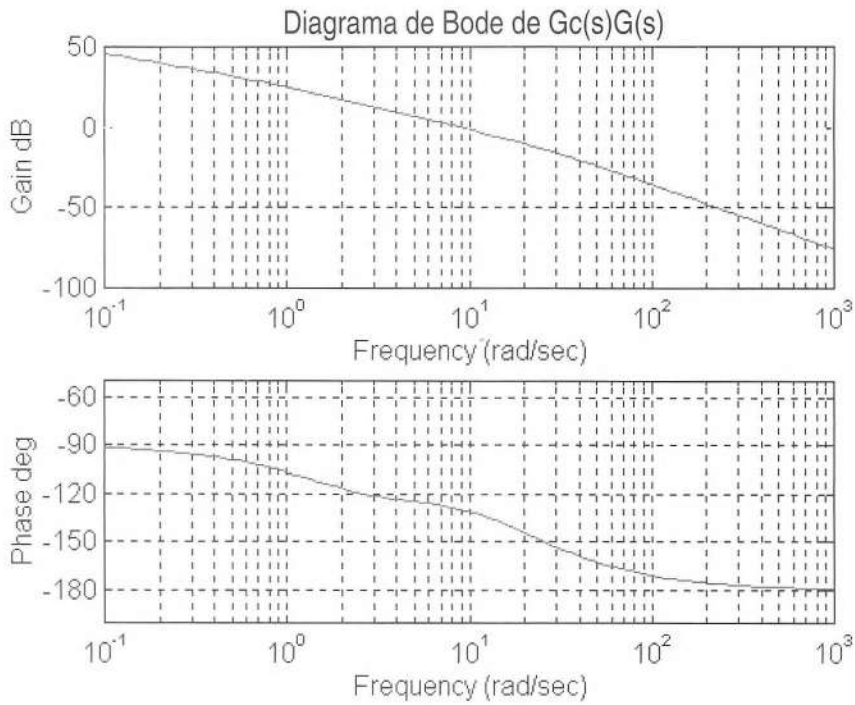


Figura 6-59

Programa en MATLAB 6-36

```
%-------Diagramas de Nyquist-------

%*****Se dibujarán los diagramas de Nyquist de G(jw), G1(jw) y Gc(jw)G(jw)*****
```

```
num1=[0 0 4];
den1=[1 2 0];
num2=[0 0 40];
den2=[1 2 0];
num3=[0 0 166.668 734.839];
den3=[1 20.371 36.742 0];
v=[-5 5 -5 5];
axis(v);axis('square')
[re1,im1,w]=nyquist(num1,den1);

%Warning: Divide by zero

z1=re1+i*im1;
r1=abs(z1);
theta1=angle(z1);
polar(theta1,r1)
grid
hold

%Current plot held

[re2,im2,w]=nyquist(num2,den2);

%Warning: Divide by zero

z2=re2+i*im2;
r2=abs(z2);
theta2=angle(z2);
polar(theta2,r2)
[re3,im3,w]=nyquist(num3,den3);

%Warning: Divide by zero

z3=re3+i*im3;
r3=abs(z3);
theta3=angle(z3);
polar(theta3,r3)
title('Diagramas de Nyquist de G(jw), G1(jw) y Gc(jw)G(jw)')
text(-1.9,3,'G(jw)')
text(-2.35,5.6,'Gc(jw)G(jw)')
text(-2.8,3.5,'G1(jw)')

hold

%Current plot released

%*****Restaure la perspectiva gráfica normal*****

axis('normal')
```

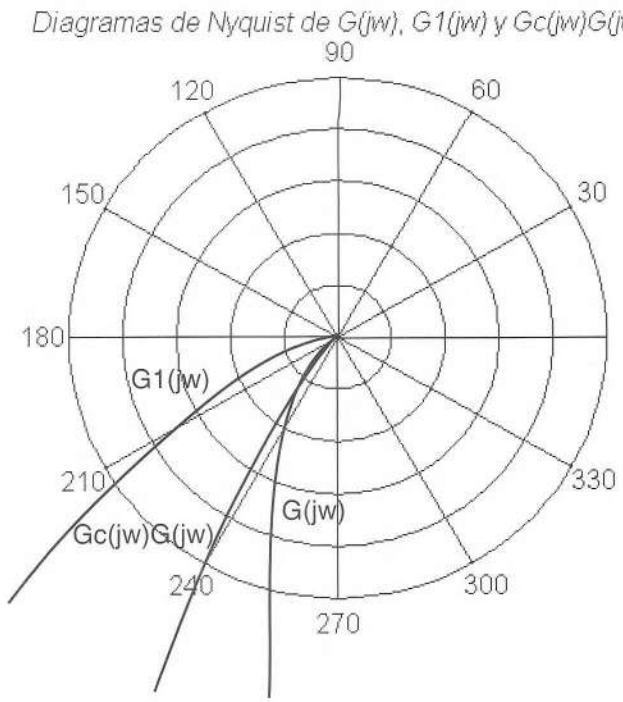


Figura 6-60

Programa en MATLAB 6-37

```
%-------Respuestas a un salto unitario-------

%*****En este programa se obtendrán las respuestas a un salto unitario del
%sistema sin compensar y del sistema compensado*****

num1=[0 0 4];
den1=[1 2 4];
num2=[0 0 166.668 734.839];
den2=[1 20.371 203.41 734.839];
v=[0 6 0 1.6];
axis(v);
t=0:0.05:6;
c1=step(num1,den1,t);
c2=step(num2,den2,t);
plot(t,c1,t,c2)
grid
title('Respuestas a un salto unitario de los sistemas sin compensar y compen-
sado')
xlabel('t Seg')
ylabel('Salidas c1 y c2')
text(2.5,1.1,'Sistema sin compensar')
text(0.3,1.25,'Sistema compensado')
```

Respuesta a un salto unitario. Comprobaremos la respuesta a un salto unitario del sistema diseñado. Representaremos tanto la respuesta a un salto unitario del sistema diseñado como la del sistema original sin compensar.

La función de transferencia en lazo cerrado del sistema original sin compensar es

$$\frac{C_1(s)}{R_1(s)} = \frac{4}{s^2 + 2s + 4}$$

La función de transferencia en lazo cerrado del sistema compensado es

$$\frac{C_2(s)}{R_2(s)} = \frac{41{,}667(s + 4{,}409) \times 4}{(s + 18{,}371)s(s + 2) + 41{,}667(s + 4{,}409) \times 4}$$
$$= \frac{166{,}668s + 734{,}839}{s^3 + 20{,}371s^2 + 203{,}41s + 734{,}839}$$

El programa de MATLAB 6-37 produce las respuestas a un salto unitario del sistema sin compensar y compensado. En la Figura 6-61 se muestran las curvas de respuesta resultantes.

Respuesta a una rampa unitaria. Es conveniente verificar la respuesta a una rampa unitaria del sistema compensado. Como $K_v = 20$ seg^{-1}, el error en estado estacionario para una entrada en rampa unitaria será $1/K_v = 0.05$. La constante de error de velocidad estática del sistema sin compensar es 2 seg-1. Por tanto el sistema sin compensar original tendrá un gran error en estado estacionario para una entrada en rampa unitaria.

El programa de MATLAB 6-38 genera las curvas de respuesta a una entrada en rampa unitaria. [Observe que la curva de respuesta a una entrada en rampa unitaria se obtiene como la respuesta a un salto unitario de $C_i(s)/sR(s)$, donde i =1,2]. En la Figura 6-62 se muestran las curvas resultantes. El sistema compensado tiene un error en estado estacionario igual a una décima parte del que tiene el sistema original sin compensar.

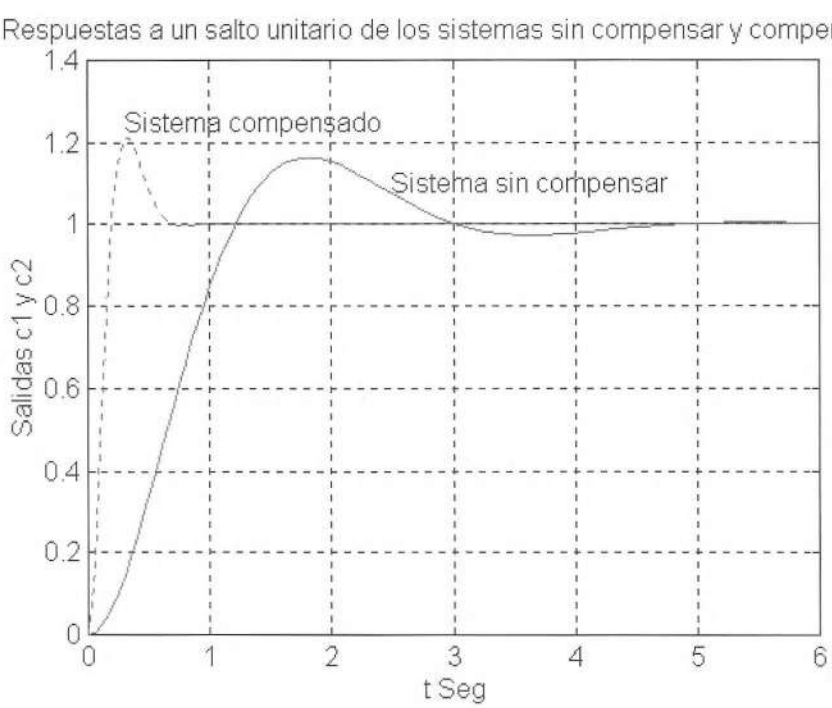


Figura 6-61

Programa en MATLAB 6-38

```
%-------Respuestas a una rampa unitaria-------

%*****En este programa se obtendrán las respuestas a una rampa unitaria del
%sistema sin compensar y del sistema compensado*****

num1=[0 0 0 4];
den1=[1 2 4 0];
num2=[0 0 0 166.668 734.839];
den2=[1 20.371 203.41 734.839 0];
t=0:0.1:4;
c1=step(num1,den1,t);
c2=step(num2,den2,t);
plot(t,c1,t,c2,t,t,'-')
grid
title('Respuestas a una rampa unitaria de los sistemas sin compensar y compen-
sado')
xlabel('t Seg')
ylabel('Salidas c1 y c2')
text(2.5,1.75,'Sistema')
text(2.5,1.6,'sin compensar')
text(1.65,2.5,'Sistema')
text(1.65,2.35,'compensado')
```

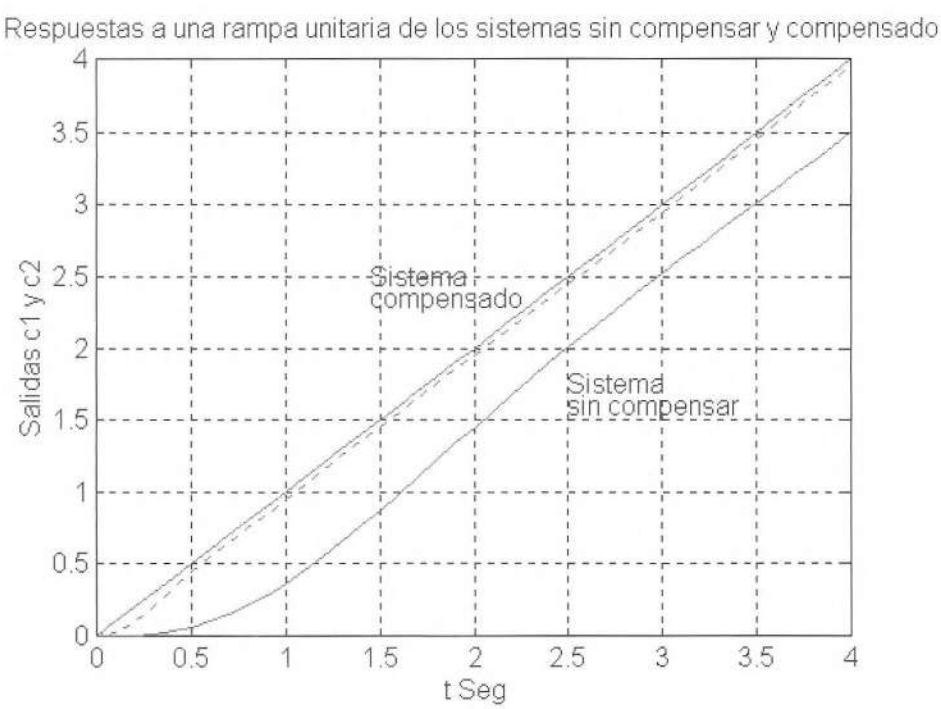


Figura 6-62

EJEMPLO 6-29: DISEÑO DE UN COMPENSADOR DE RETRASO-ADELANTO

Sea el sistema de realimentación unitaria cuya función de transferencia en lazo abierto es

$$G(s) = \frac{K}{s(s+1)(s+4)}$$

Diseñar un compensador $G_c(s)$ tal que la constante de error de velocidad estática es 10 seg^{-1}, el margen de fase de 50° y el margen de ganancia de 10 dB o más

Diseñaremos un compensador de retraso-adelanto de la forma

$$G_c(s) = K_c \frac{\left(s+\frac{1}{T_1}\right)\left(s+\frac{1}{T_2}\right)}{\left(s+\frac{\beta}{T_1}\right)\left(s+\frac{1}{\beta T_2}\right)}$$

La función de transferencia en lazo abierto del sistema compensado es $G_c(s)G(s)$. Como la ganancia K de la planta es ajustable, asumimos que $K_c = 1$. Entonces $\lim_{s\to 0} G_c(s) = 1$. Del requerimiento sobre la constante de error de velocidad estática obtenemos

$$K_v = \lim_{s\to 0} sG_c(s)G(s) = \lim_{s\to 0} sG_c(s)\frac{K}{s(s+1)(s+4)}$$

$$= \frac{K}{4} = 10$$

Por tanto

$$K = 40$$

Primero representaremos un diagrama de Bode del sistema no compensado con $K = 40$. El programa de MATLAB 6-39 se puede utilizar para representar este diagrama de Bode. En la Figura 6-63 se muestra el diagrama obtenido.

De la Figura 6-63, se encuentra que el margen de fase del sistema sin compensar es de -16°, lo que indica que este sistema es inestable. El siguiente paso en el diseño del compensador de retraso-adelanto es elegir una nueva frecuencia de cruce de la ganancia. De la curva de ángulo de fase de $G(jw)$, se observa que la frecuencia de cruce de la fase es w = 2 rad/seg de forma que el ángulo de adelanto de fase requerido en w = 2 rad/seg es de unos 50°. Un simple compensador de retraso-adelanto puede proporcionar esta cantidad de adelanto en el ángulo de fase de forma sencilla.

Una vez que escogemos que la frecuencia de la ganancia de cruce es 2 rad/seg, podemos determinar las frecuencias esquinas de la parte de retraso de fase del compensador de retraso-adelanto. Elegimos la frecuencia esquina $w = 1/T_2$ (que corresponde al cero de la parte de retraso de fase del compensador) para que esté una década por debajo de la nueva frecuencia de cruce de la ganancia, es decir en w = 0.2 rad/seg. Para la otra frecuencia esquina $w = 1/(\beta T_2)$, necesitamos el valor de β. El valor de β se puede determinar a partir de la consideración de la parte de adelanto del compensador, tal como se muestra a continuación.

Programa en MATLAB 6-39

```
num=[0 0 0 40]; den=[1 5 4 0];
w=logspace(-1,1,100);
bode(num,den,w),title('Diagrama de Bode de G(s)=40/[s(s+1)(s+4)]')
```

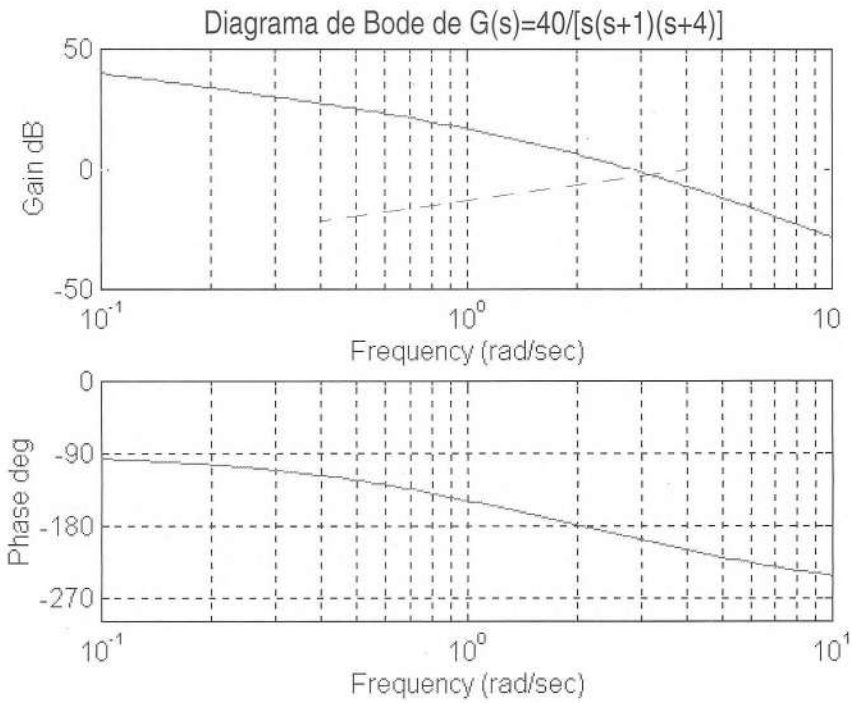


Figura 6-63

Para el compensador de adelanto, el máximo ángulo de adelanto de fase ϕ_m viene dado por

$$\operatorname{sen}\phi_m = \frac{\beta - 1}{\beta + 1}$$

Obsérvese que $\beta = 10$ corresponde a $\phi_m = 54.9°$. Como necesitamos un margen de fase de 50°, podemos escoger $\beta = 10$. (Advierta que estamos utilizando algunos grados menos que el máximo ángulo 54.9°). Así

$$\beta = 10$$

Entonces la frecuencia esquina $w = 1/(\beta T_2)$. (Que corresponde al polo de la parte de retraso de fase del compensador) se hace

$$\omega = 0{,}02$$

La función de transferencia de la parte de retraso de fase del compensador de retraso-adelanto es

$$\frac{s + 0{,}2}{s + 0{,}02} = 10\left(\frac{5s + 1}{50s + 1}\right)$$

La parte de adelanto de fase se puede determinar como sigue: como la nueva frecuencia de cruce de la ganancia es w = 2 rad/seg, de la Figura 6-63, $|G(j2)|$ se encuentra que es 6 dB. Por tanto, si el compensador de retraso-adelanto contribuye con -6 dB en w = 2 rad/seg, entonces la nueva frecuencia de cruce de la ganancia es como se desea. De este requisito, es posible dibujar una línea recta de pendiente 20 dB/década que pasa a través del punto (-6 dB, 2 rad/seg). (Tal línea se ha dibujado manualmente en la Figura 6-63). La intersección de esta línea y la línea de 0 dB y la línea de -20 dB determina las frecuencias esquinas. De esta consideración, las frecuencias esquinas para la parte de adelanto se pueden determinar como w = 0.4 rad/seg y w = 4 rad/seg. Así la función de transferencia de la parte de adelanto del compensador de retraso-adelanto es

$$\frac{s+0,4}{s+4} = \frac{1}{10}\left(\frac{2,5s+1}{0,25s+1}\right)$$

Combinando las funciones de transferencia de ambas partes del compensador de retraso y adelanto, podemos obtener la funcion de transferencia $G_c(s)$ del compensador de retraso-adelanto. Como elegimos $K_v = 1$ tenemos

$$G_c(s) = \frac{s+0,4}{s+4}\ \frac{s+0.2}{s+0,02} = \frac{(2,5s+1)(5s+1)}{(0,25s+1)(50s+1)}$$

El diagrama de Bode del compensador de retraso-adelanto $G_c(s)$ se puede obtener introduciendo en el computador el programa de MATLAB 6-40. En la Figura 6-64 se muestra el diagrama resultante.

Programa en MATLAB 6-40

```
num=[1 0.6 0.08];
den=[1 4.02 0.08];
bode(num,den)
title('Diagrama de Bode de un compensador de adelanto-retraso')
```

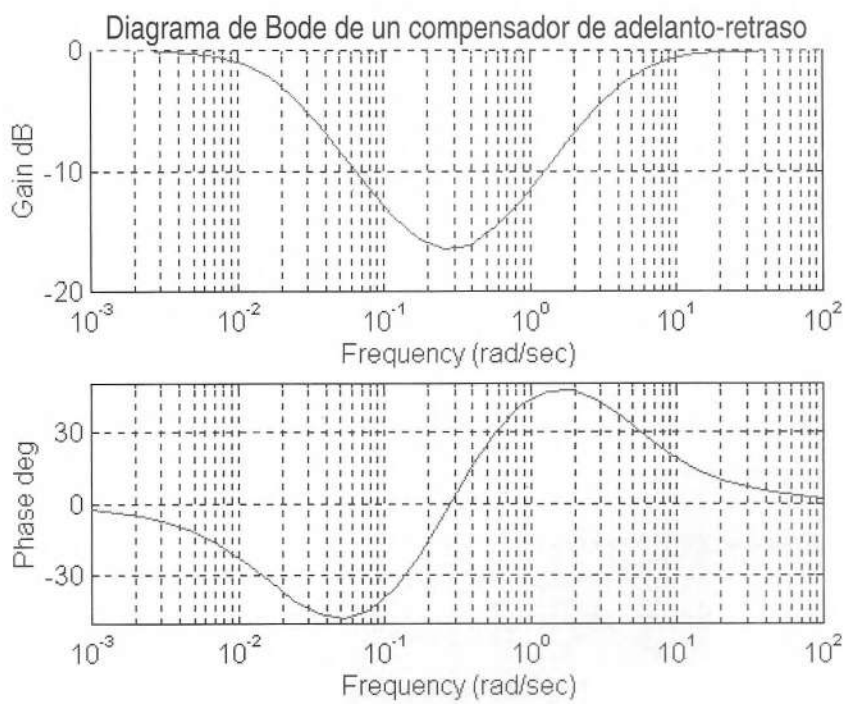


Figura 6-64

La función de transferencia en lazo abierto del sistema compensado es

$$G_c(s)G(s) = \frac{(s+0,4)(s+0,2)}{(s+4)(s+0,02)}\ \frac{40}{s(s+1)(s+4)}$$

$$= \frac{40s^2 + 24s + 3{,}2}{s^5 + 9{,}02s^4 + 24{,}18s^3 + 16{,}48s^2 + 0{,}32}$$

Las curvas de magnitud y ángulo de fase de la función de transferencia en lazo abierto diseñada $G_c(s)G(s)$ se muestran en el diagrama de Bode de la Figura 6-65. Esste diagrama se obtiene utilizando el programa de MATLAB 6-41. Obsérvese que el polinomio del denominador den fue obtenido utilizando la orden conv, tal como sigue:

```
a=[1 4.02 0.08];
b=[1 5 4 0];
conv(a,b)

ans =

    1.0000    9.0200   24.1800   16.4800    0.3200         0
```

Programa en MATLAB 6-41

```
num=[0 0 0 40 24 3.2];
den=[1 9.02 24.18 16.48 0.32 0];
bode(num,den)
title('Diagrama de Bode de Gc(s)G(s)')
```

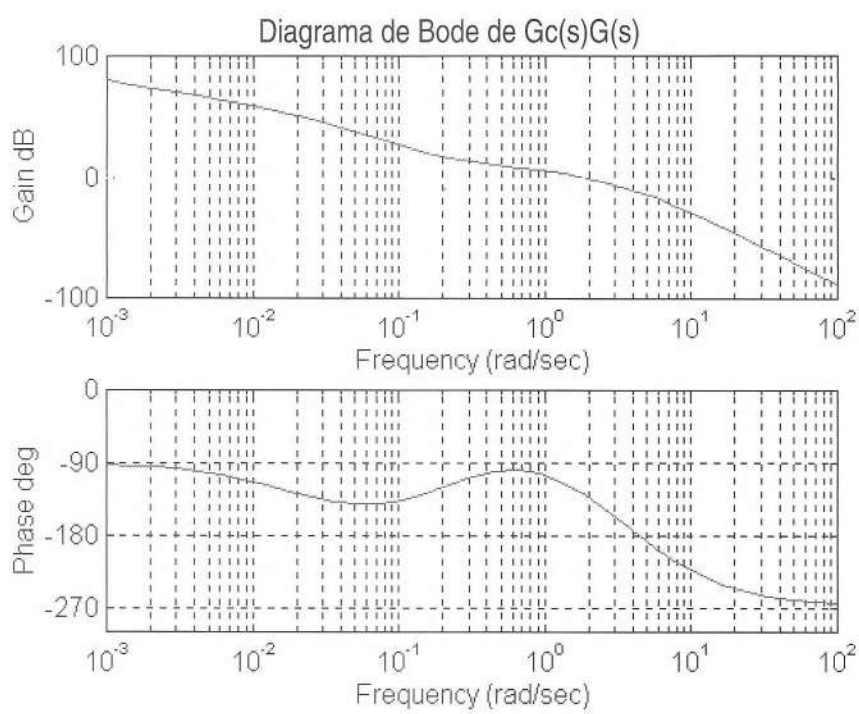


Figura 6-65

De la Figura 6-65, vemos que la frecuencia de cruce de la ganancia está ligeramente desplazada de 2 rad/seg a un valor más pequeño. La frecuencia de cruce de la ganancia se puede encontrar representando el diagrama de Bode en la región $1 \leq w \leq 10$. Se encuentra que es $w = 1.86$ rad/seg. [Este pequeño desplazamiento del valor de la frecuencia de cruce de la ganancia de su valor supuesto (2 rad/seg en este caso) siempre ocurre con este método de diseño].

El programa de MATLAB 6-42 dibujara el diagrama de Nyquist sobre un plano polar. En la Figura 6-66 se muestra el diagrama polar resultante.

Programa en MATLAB 6-42

```
num=[0 0 0 40 24 3.2];
den=[1 9.02 24.18 16.48 0.32 0];
w=0.1:0.01:1.86;
v=[-2 2 -2 2];
axis(v);axis('square')
[re,im,w]=nyquist(num,den,w);
z=re+i*im;
r=abs(z);
theta=angle(z);
polar(theta,r)
grid
title('Comprueba el margen de fase');
text(-1.75,1.8,'Diagrama de Nyquist')
text(-3,1.5,'El margen de fase')
text(-2.9,1.6,'son 50 grados.')

%*****Restaure la perspectiva gráfica normal*****

axis('normal')
```

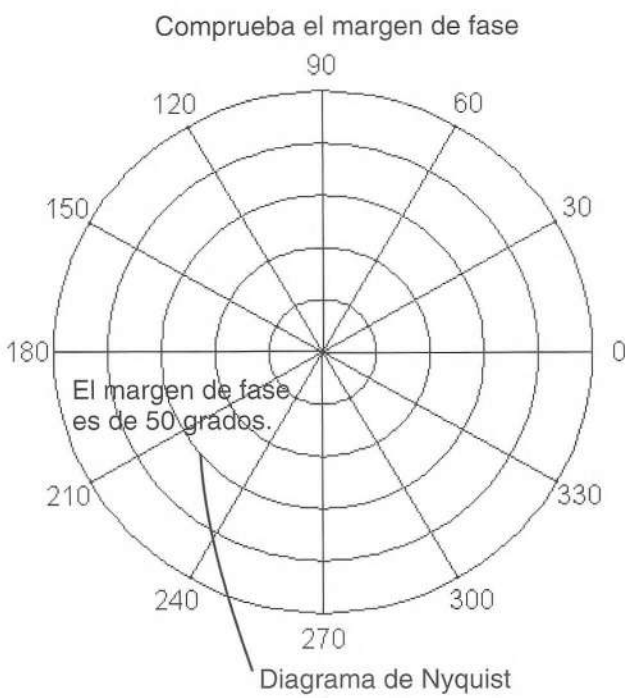


Figura 6-66

Como el margen de fase del sistema compensado es de 50°, el margen de ganancia de 12.5 dB y la constante de error de velocidad estáticaes 10 seg^{-1}, se cumplen todos los requerimientos.

La Figura 6-67 muestra los diagramas de Nyquist de $G(jw)$ (caso sin compensar) y $G_c(jw)G(jw)$ (caso compensado). El programa en MATLAB para obtener la Figura 6-67 se muestra en el programa 6-43.

Si se desean representar los diagramas de Bode de $G(jw)$, $G_c(jw)$ y $G_c(jw)G(jw)$ en la misma gráfica, se puede utilizar el programa de MATLAB 6-44. La Figura 6-68 muestra los diagramas producidos por este programa de MATLAB.

A continuación investigaremos las características de la respuesta transitoria del sistema diseñado.

Respuesta a un salto unitario. Observando que

$$G_c(s)G(s) = \frac{40(s+0,4)(s+0,2)}{(s+4)(s+0,02)s(s+1)(s+4)}$$

tenemos

$$\frac{C(s)}{R(s)} = \frac{G_c(s)G(s)}{1+G_c(s)G(s)}$$

$$= \frac{40(s+0,4)(s+0,2)}{(s+4)(s+0,02)s(s+1)(s+4)+40(s+0,4)(s+0,2)}$$

Para determinar el polinomio del denominador con MATLAB, podemos proceder como sigue. Definimos

$$a(s) = (s+4)(s+0,02) = s^2+4,02s+0,08$$

$$b(s) = s(s+1)(s+4) = s^3+5s^2+4s$$

$$c(s) = 40(s+0,4)(s+0,2) = 40s^2+24s+3,2$$

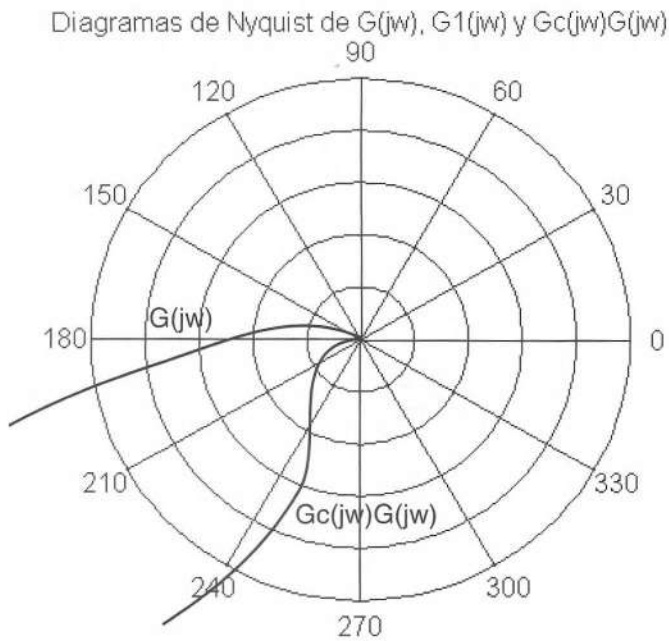


Figura 6-67

Programa en MATLAB 6-43

```
num1=[0 0 0 40];
den1=[1 5 4 0];
num2=[0 0 0 40 24 3.2];
den2=[1 9.02 24.18 16.48 0.32 0];
v=[-5 5 -5 5];
axis(v);axis('square')
[re1,im1,w]=nyquist(num1,den1);

%Warning: Divide by zero

z1=re1+i*im1;
r1=abs(z1);
theta1=angle(z1);
polar(theta1,r1)
grid
hold

%Current plot held

[re2,im2,w]=nyquist(num2,den2);

%Warning: Divide by zero

z2=re2+i*im2;
r2=abs(z2);
theta2=angle(z2);
polar(theta2,r2)
title('Diagramas de Nyquist de G(jw) y Gc(jw)G(jw)');
text(3.14,3.2,'G(jw)')
text(-1.6,3.5,'Gc(jw)G(jw)')

hold

%Current plot released

%*****Restaure la perspectiva gráfica normal*****

axis('normal')
```

Entonces tenemos

$$a = [1 \quad 4.02 \quad 0.08]$$
$$b = [1 \quad 5 \quad 4 \quad 0]$$
$$c = [40 \quad 24 \quad 3.2]$$

Programa en MATLAB 6-44

```
%-------Diagramas de Bode-------

%*****En este programa se dibujarán los diagramas de Bode de
%G(w), Gc(w) y Gc(w)G(w)*****

num=[0 0 0 40];
den=[1 5 4 0];
numc=[1 0.6 0.08];
denc=[1 4.02 0.08];
w=logspace(-3,2,100);
[mag,fase,w]=bode(num,den,w);
magdB=20*log10(mag);
[magc,fasec,w]=bode(numc,denc,w);
magcdB=20*log10(magc);

%*****Defina la magnitud de Gc(w)G(w) en decibelios como mgdB.
%mgdB se obtiene de mgdB=magdB+magcdB*****

mgdB=magdB+magcdB;

%*****Primero se dibujarán las curvas de magnitud*****

semilogx(w,magdB,'o',w,magdB,'-',w,magcdB,'x',w,magcdB,'-',...
w,mgdB,'--')
grid
title('Diagramas de bode de G(w), Gc(w) y Gc(w)G(w)')
xlabel('Frecuencia (rad/seg)')
ylabel('Ganancia dB')
text(0.1,40,'G(w)')
text(0.1,-25,'Gc(w)')
text(0.03,12,'Gc(w)G(w)')

%*****A continuación se dibujarán las curvas de fase*****

%*****Defina la fase de Gc(w)G(w) como fa. fa se obtiene de fa=fase+fasec*****

fa=fase+fasec;

%*****Dibuje la línea de -180 grados introduciendo la siguiente sentencia*****

fa180=-180*ones(1,100);

%*****Introduzca la siguiente orden de dibujo*****

semilogx(w,fase,'o',w,fase,'-',w,fasec,'x',w,fasec,'-',...
w,fa,'--',w,fa180,'-')
grid
xlabel('Frecuencia (rad/seg)')
ylabel('Fase grados')
text(0.01,-85,'G(w)')
text(0.01,-20,'Gc(w)')
text(0.01,-160,'Gc(w)G(w)')
```

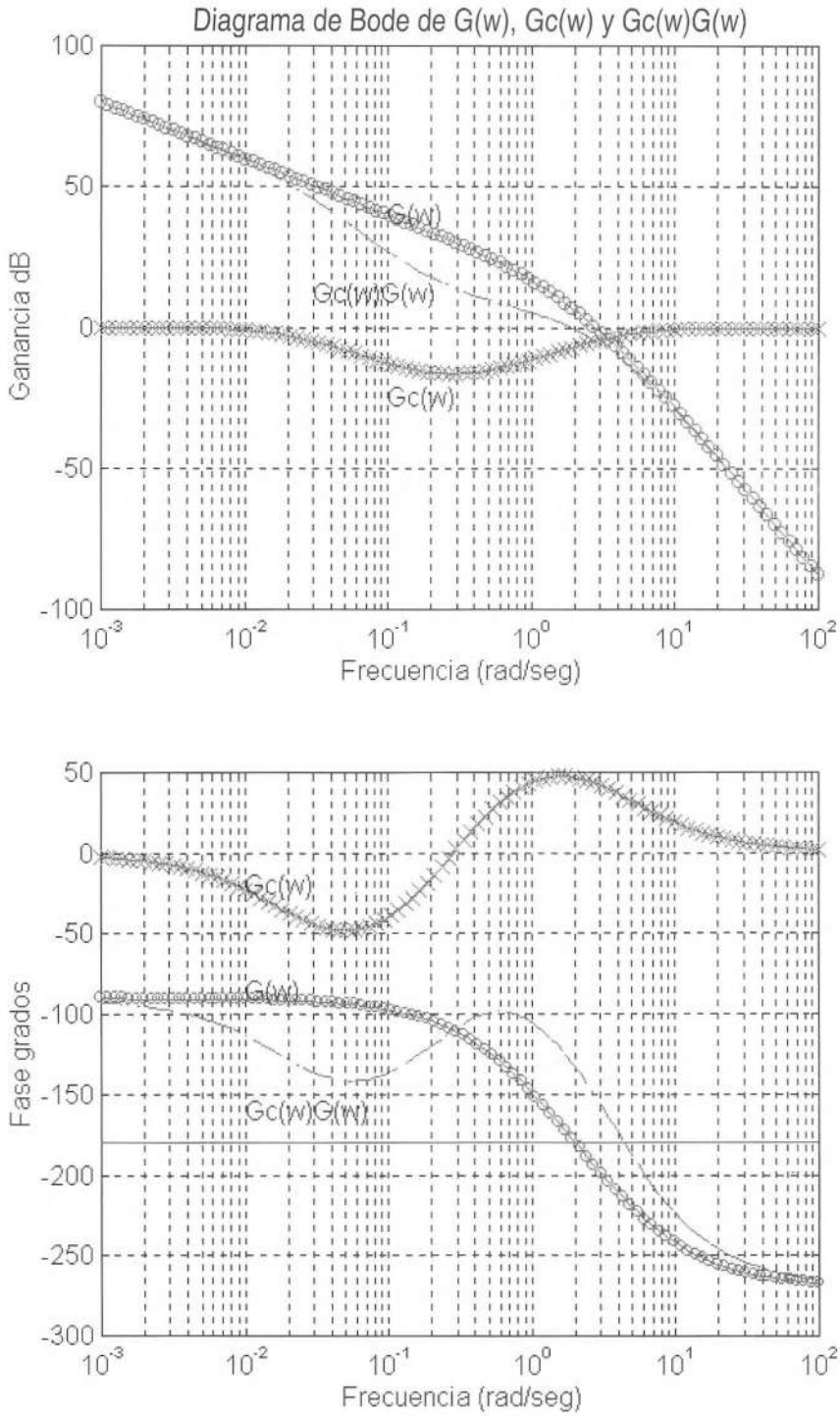


Diagrama de Bode de G(w), Gc(w) y Gc(w)G(w)
Ganancia dB
100
50
0
-50
-100
G(w)
Gc(w)G(w)
Gc(w)
10^{-3}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
Frecuencia (rad/seg)
Fase grados
50
0
-50
-100
-150
-200
-250
-300
Gc(w)
G(w)
Gc(w)G(w)
10^{-3}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
Frecuencia (rad/seg)

Figura 6-68

Utilizando el siguiente programa en MATLAB, obtenemos el polinomio del denominador.

```
a=[1 4.02 0.08];
b=[1 5 4 0];
c=[40 24 3.2];
p=[conv(a,b)]+[0 0 0 c]

p =

    1.0000    9.0200   24.1800   56.4800   24.3200    3.2000
```

Programa en MATLAB 6-45

```
%-------Respuesta a un salto unitario-------

num=[0 0 0 40 24 3.2];
den=[1 9.02 24.18 56.48 24.32 3.2];
t=0:0.2:40;
v=[0 40 0 1.4];
axis(v);
step(num,den,t)
grid
title('Respuesta a un salto unitario del sistema diseñado')
```

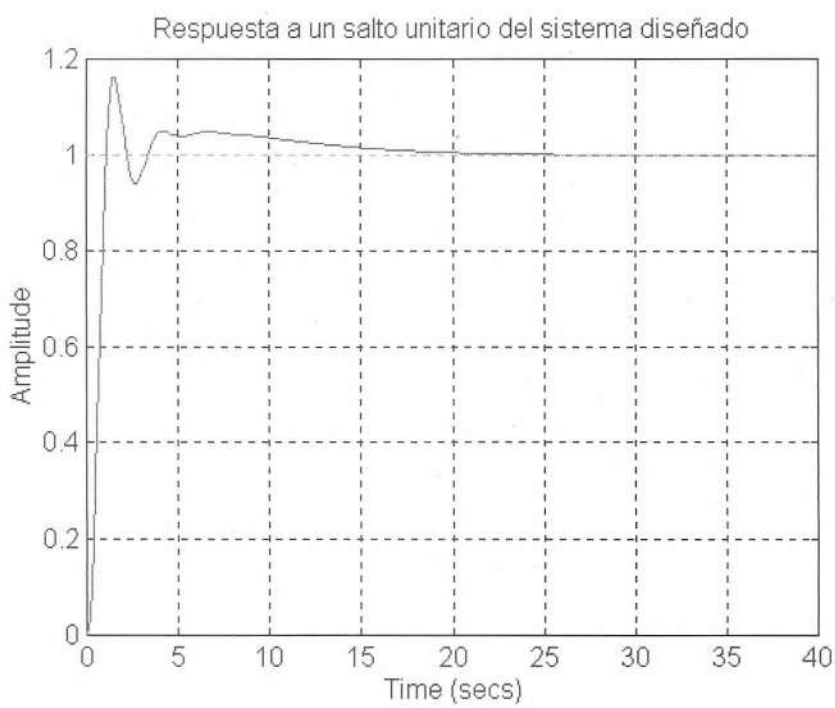


Figura 6-69

Para obtener la respuesta a un salto unitario, podemos utilizar el programa de MATLAB 6-45, que produce la curva de respuesta a un salto unitario que se muestra en la Figura 6-69.

Respuesta a una rampa unitaria. La respuesta a una rampa unitaria de este sistema se puede obtener introduciendo en el computador el programa de MATLAB 6-46. Aquí convertimos la respuesta a una rampa unitaria de $G_cG/(1 + G_cG)$ en la respuesta a un salto unitario de $G_cG/[s(1 + G_cG)]$. En la Figura 6-70 se muestra la curva de respuesta a una rampa unitaria obtenida.

Programa en MATLAB 6-46

```
%-------Respuesta a una rampa unitaria-------

num=[0 0 0 0 40 24 3.2];
den=[1 9.02 24.18 56.48 24.32 3.2 0];
t=0:0.2:20;
c=step(num,den,t);
plot(t,c,t,t,'--')
grid
title('Respuesta a una rampa unitaria del sistema diseñado')
xlabel('Tiempo (seg)')
ylabel('Amplitud')
```

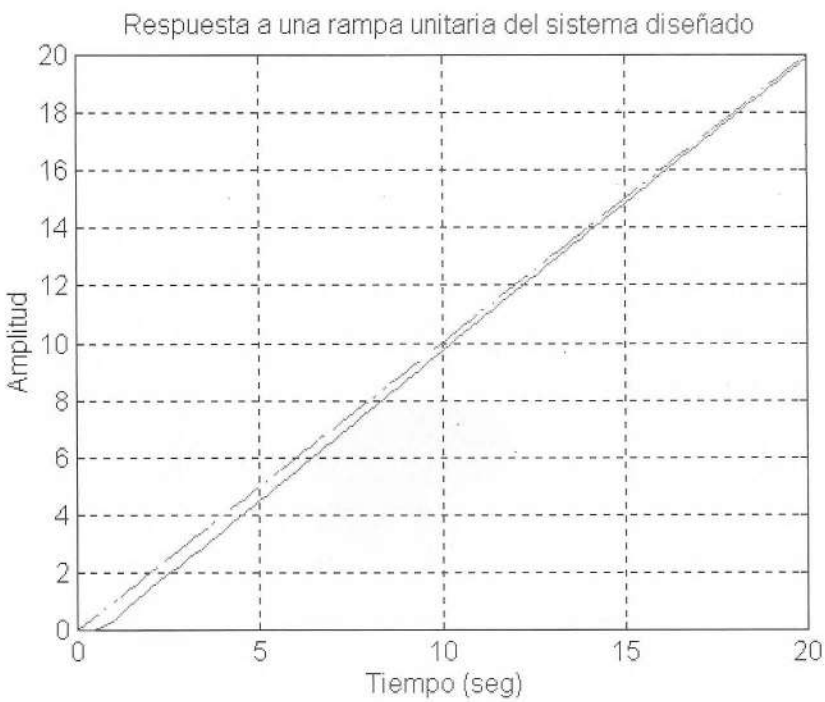


Figura 6-70

6.6 RESPUESTA EN FRECUENCIA DE SISTEMAS DE CONTROL EN TIEMPO DISCRETO

La transformada z, transforma las bandas primarias y complementarias del semiplano izquierdo s en el círculo unidad del plano z. Como los métodos de respuesta en frecuencia convencional tratan con el semiplano izquierdo completo, no se pueden utilizar con el plano z.

Sin embargo esta dificultad se puede superar transformando la función de transferencia discreta del plano z en el plano w. La transformación, normalmente conocida como transformación w o transformación bilineal se define de la forma siguiente

$$z = \frac{1 + (T/2)w}{1 - (T/2)w} \tag{6-15}$$

donde T es el periodo de muestreo utilizado en el sistema de control en tiempo discreto que está bajo consideración. Al convertir una función de transferencia discreta dada en el plano z en una función racional de w, se pueden extender los métodos de respuesta en frecuencia al caso de los sistemas de control en tiempo discreto. Si se resuelve (6-5) para w, se obtiene la relación inversa

$$w = \frac{2}{T}\frac{z-1}{z+1} \tag{6-16}$$

Mediante las transformaciones z y w, la banda primaria del semiplano izquierdo s se transforma en primer lugar en el interior del círculo unidad en el plano z y a continuación en el semiplano izquierdo completo del plano w. En la Figura 6-71 se representa el proceso de estas dos transformaciones. (Observe que en el plano s consideramos solo la banda primaria). También conviene darse cuenta que el origen del plano z se transforma en el punto w = -2/T del plano w y que cuando s varía desde 0 a jw/2, a lo largo del eje jw en el plano s, z varía de -1 a 1 a lo largo del círculo unidad en el plano z y w va de 0 a ∞ sobre el eje imaginario en el plano w.

Aunque el semipalno izquierdo del plano w corresponde al semiplano izquierdo del plano s, y el eje imaginario del plano w también corresponde al eje imaginario del plano s, hay diferencias entre los dos planos. La diferencia principal es que la conducta en el plano s en el rango de frecuencias $-0.5w_s \leq w \leq 0.5w_s$ se transforma en el el rango $-\infty < v < \infty$, donde v es la frecuencia ficticia en el plano w.

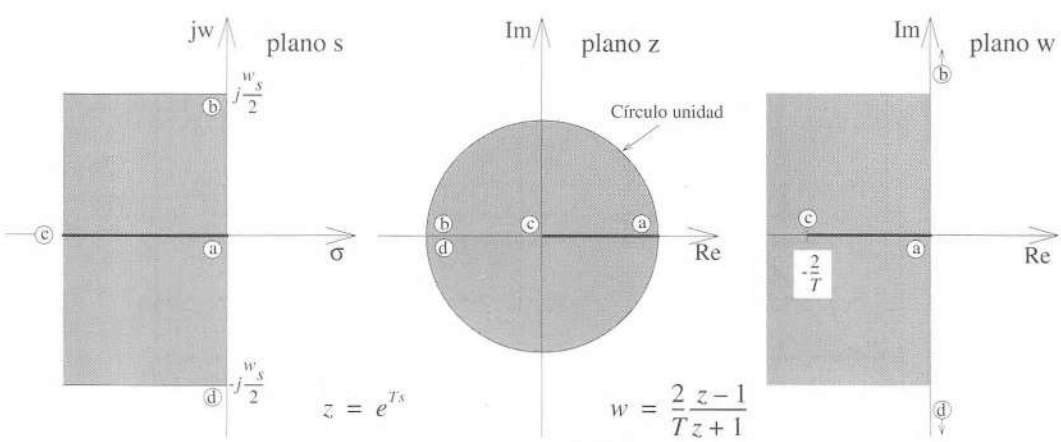


Figura 6-71

Una vez que la función de transferencia discreta $G(z)$ se transforma en $G(w)$ mediante la transformación w, se puede tratar como una función de transferencia convencional en w. Al reemplazar w por jv, se pueden emplear las técnicas usuales de respuesta en frecuencia para dibujar los diagramas de Bode para la función de transferencia en w. Como $G(w)$ es una función racional en e, se puede aplicar el criterio de estabilidad de Nyquist a $G(w)$. De esta forma los conceptos de márgenes de fase y de gancia también se aplican a $G(w)$.

Aunque geométricamente el plano w se asemeja al plano s, el eje de frecuencias en el plano w se distorsiona. La frecuencia ficticia ν y la frecuencia real w se relacionan como sigue:

$$w|_{w=jv} = jv = \frac{2}{T}\left.\frac{z-1}{z+1}\right|_{z=e^{j\omega T}} = \frac{2}{T}\frac{e^{j\omega T}-1}{e^{j\omega T}+1}$$

$$= \frac{2}{T}\frac{e^{j(1/2)\omega T}-e^{-j(1/2)\omega T}}{e^{j(1/2)\omega T}+e^{-j(1/2)\omega T}} = \frac{2}{T}\,j\tan\frac{\omega T}{2}$$

o

$$v = \frac{2}{T}\tan\frac{\omega T}{2} \tag{6-17}$$

La ecuación (6-17) da la relación entre la frecuencia real w y la frecuencia ficticia ν. Observe que cuando la frecuencia real w se mueve desde -0.5w_s a 0, la frecuencia ficticia ν va desde -∞ a 0 y cuando w se mueve desde 0 hasta 0.5w_s, ν lo hace desde 0 a ∞.

Teniendo en cuenta (6-17), la frecuencia real w se puede traducir en la frecuencia ficticia ν. Por ejemplo, *si el ancho de banda se especifica como* w_b, *entonces el correspondiente ancho de banda en el plano w es (2/T)* tan (w_bT/2). La Figura 6-72 muestra la relación entre la frecuencia ficticia ν π por 0.5T veces y la frecuencia real w para el rango de frecuencias entre 0 y 0.5w_s.

Obsérvese en (6-17) que si wT es pequeño entonces

$$v \cong \omega$$

Esto significa que para pequeños wT las funciones de transferencia $G(s)$ y $G(w)$ se asemejan la una a la otra. Esto es el resultado directo de la inclusión del factor de escala 2/T en (6-16). La presencia en la transformación de este factor de escala nos permite mantener las mismas constantes de error antes y después de la transformación w. (Esto significa que la función de transferencia en el plano w se aproximará a la del plano s cuando T tiende a cero. Véase el Ejemplo 6-30).

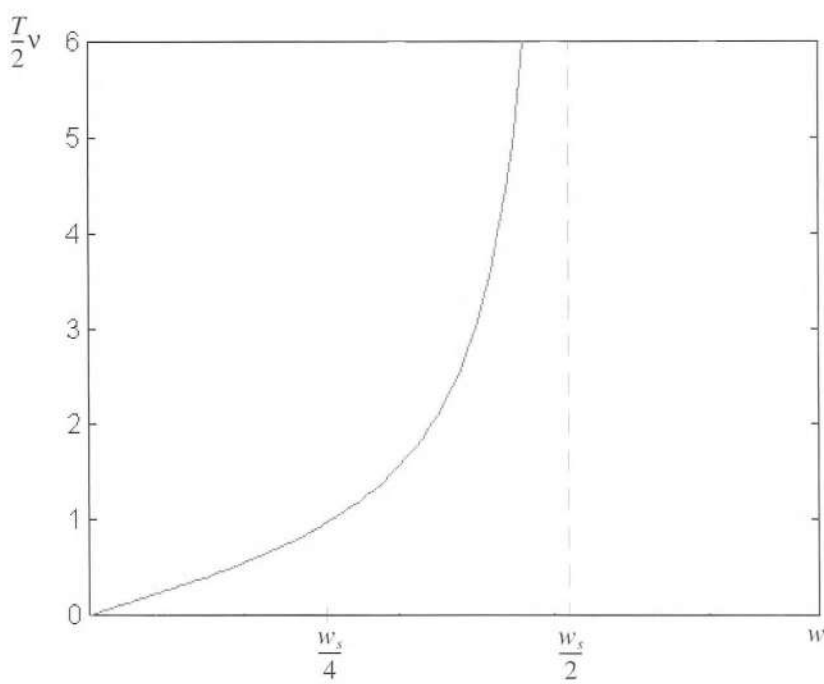


Figura 6-72

Algunos comentarios en relación con los tests de respuesta en frecuencia en los sistemas de tiempo discreto

Al efectuar los tests de respuesta en frecuencia sobre sistemas en tiempo discreto, es importante que el sistema tenga un filtro paso baja antes del muestreador de forma que se filtren las bandas laterales. En estas condiciones la respuesta del sistema lineal e invariante en el tiempo a una entrada sinusoidal preserva la frecuencia y modifica sólo la amplitud y la fase de la señal de entrada. Así la amplitud y la fase son las dos únicas cantidades que se deben de tratar.

EJEMPLO 6-30

Sea la función de transferencia del sistema que se muestra en la Figura 6-73. El periodo de muestreo T se supone que es de 0.1 seg. Obtener $G(w)$.

La transformada z de $G(s)$ es

$$G(z) = \mathrm{Z}\left[\frac{1-e^{-Ts}}{s}\,\frac{10}{s+10}\right]$$

$$= (1-z^{-1})\mathrm{Z}\left[\frac{10}{s(s+10)}\right]$$

$$= \frac{0{,}6321}{z-0{,}3679}$$

Mediante el uso de la transformación bilineal dada por (6-5) o

$$z = \frac{1+(T/2)w}{1-(T/2)w} = \frac{1+0{,}05w}{1-0{,}05w}$$

$G(z)$ se puede transformar en $G(w)$ como sigue:

$$G(w) = \frac{0{,}6321}{\dfrac{1+0{,}05w}{1-0{,}05w}-0{,}3679} = \frac{0{,}6321(1-0{,}05w)}{0{,}6321+0{,}06840w}$$

$$= 9{,}241\,\frac{1-0{,}05w}{w+9{,}241}$$

Observe que la localización del polo de la planta es s = -10 y que la del polo en el plano w es w = -9.241. El valor de la ganancia en el plano s es 10 y en el plano w es 9.241. (Así pues, ambos valores, el de la localización del polo y el de la ganancia son similares en ambos planos). No obstante, $G(w)$ tiene un cero en w =2/T = 20, aunque la planta no tiene ningún cero. Cuando el periodo de muestreo T se hace más pequeño, el cero del plano w en w = 2/T se aproxima a infinito en el semiplano derecho del plano w. Observe que tenemos que

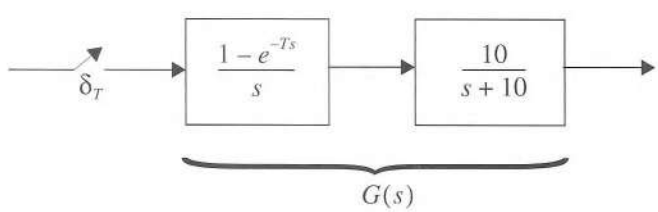


Figura 6-73

$$\lim_{w \to 0} G(w) = \lim_{s \to 0} \frac{10}{s + 10}$$

Este hecho es muy útil para comprobar los cálculos numéricos cuando se transforma $G(s)$ en $G(w)$.

G(w) es una función de transferencia de fase no mínima

Como se ha visto en el el Ejemplo 6-30, en general, uno o más ceros de $G(w)$ yacen en el semiplano derecho del plano w. La presencia de uno o más ceros en el semiplano derecho del plano w significa que $G(w)$ es una función de transferencia de fase no mínima. Por tanto, debemos ser cuidadoso al dibujar la curva del ángulo de fase en el diagrama de Bode.

Algunos comentarios sobre el problema de cuantificación de los coeficientes

Desde el punto de vista de la implementación en microprocesador de los compensadores de adelanto de fase, retardo de fase y adelanto-retardo de fase, los compensadores de adelanto de fase no presentan problemas de cuantificación de los coeficientes, porque la localización de los polos y de los ceros están suficientemente separados. Sin embargo, en el caso de compensadores por retardo de fase y por retardo-adelanto de fase, la red de retardo de fase presenta un problema de cuantificación de coeficientes porque la localización de los polos y de los ceros están más próximos unos a otros. (Están cerca del punto z = 1).

Como los coeficientes del filtro se pueden realizar con palabras binarias que utilizan un número limitado de bits, si este número de bits es insuficiente, las localizaciones de los polos y de los ceros del filtro puede que no se realice de la forma que se desea y que el filtro resultante no se comporte como se espera.

Puesto que pequeñas desviaciones en las localizaciones de los polos y de los ceros de sus posiciones deseadas pueden tener un efecto importante sobre las características de la respuesta en frecuencia del compensador, su versión digital puede no funcionar tal como se preveía. Para minimizar el efecto del problema de la cuantificación de los coeficientes, es necesario estructurar el filtro de forma que se minimicen las imprecisiones de los coeficientes debidas a la cuantificación.

Debido a que la sensibilidad de las raíces de los polinomios frente a variaciones en los parámetros se hace severa cuando aumenta el orden del polinomio, no resulta deseable implementar una realización directa de un filtro de orden elevado. Es preferible colocar elementos de bajo orden en cascada o en paralelo. Como una cuestión práctica, si en principio escogemos los polos y los ceros del compensador digital en posiciones permitidas, entonces se puede evitar el problema de la cuantificación. (El punto importante que hay que recordar es que los polos y los ceros del filtro en el plano z deben permanecer en un número finito de puntos discretos permitidos.)

Transformación de *G(s)* a *G(z)*

La transformación de una función de transferencia continua $G(s)$ en una función de transferencia discreta $G(z)$ se puede realizar mediante el siguiente procedimiento. En primer lugar definir:

$$G(s) = \frac{\text{num}}{\text{den}}$$

A continuación utilizar las órdenes siguientes:

```
[A,B,C,D] = tf2ss(num,den)
[G,H] = c2d(A,B,Ts)
[numz,denz] = ss2tf(G,H,C,D)
```

donde Ts es el periodo de muestreo en segundos. Los vectores filas numz y denz especifican los coeficientes del numerador y del denominador de $G(z)$ (en potencias decrecientes de z). Así

$$G(z) = \frac{\text{numz}}{\text{denz}}$$

Para más detalles véanse los Ejemplos 6-31 y 6-32.

EJEMPLO 6-31

Suponiendo que el periodo de muestreo T es de 0.1 seg, transformar

$$G(s) = \frac{10}{s + 10}$$

en $G(z)$.

El programa de MATLAB 6-47 genera $G(z)$, la transformada z de $G(s)$ precedida por el retenedor de orden cero. Así, obtenemos

$$G(z) = \frac{0{,}6321}{z - 0{,}3679}$$

Programa en MATLAB 6-47

```
%-------Transformar G(s) a G(z)-------

%*****Introduzca el numerador y el denominador de G(s)*****

num=[0 10];
den=[1 10];

%*****El numerador (numz) y el denominador (denz) de G(z) se pueden obtener
introduciendo
%las siguientes órdenes*****

[A,B,C,D]=tf2ss(num,den);
[G,H]=c2d(A,B,0.1);
[numz,denz]=ss2tf(G,H,C,D)

numz =

         0    0.6321


denz =

    1.0000   -0.3679
```

que es correcto. Las salidas intermedias, tales como las matrices A, B, C, D, G y H se muestran a continuación:

donde Ts es el periodo de muestreo en segundos. Los vectores filas numz y denz especifican los coeficientes del numerador y del denominador de $G(z)$ (en potencias decrecientes de z). Así

$$G(z) = \frac{\text{numz}}{\text{denz}}$$

Para más detalles véanse los Ejemplos 6-31 y 6-32.

EJEMPLO 6-31

Suponiendo que el periodo de muestreo T es de 0.1 seg, transformar

$$G(s) = \frac{10}{s + 10}$$

en $G(z)$.

El programa de MATLAB 6-47 genera $G(z)$, la transformada z de $G(s)$ precedida por el retenedor de orden cero. Así, obtenemos

$$G(z) = \frac{0{,}6321}{z - 0{,}3679}$$

Programa en MATLAB 6-47

```
%-------Transformar G(s) a G(z)-------

%*****Introduzca el numerador y el denominador de G(s)*****

num=[0 10];
den=[1 10];

%*****El numerador (numz) y el denominador (denz) de G(z) se pueden obtener
introduciendo
%las siguientes órdenes*****

[A,B,C,D]=tf2ss(num,den);
[G,H]=c2d(A,B,0.1);
[numz,denz]=ss2tf(G,H,C,D)

numz =

         0    0.6321


denz =

    1.0000   -0.3679
```

que es correcto. Las salidas intermedias, tales como las matrices A, B, C, D, G y H se muestran a continuación:

donde Ts es el periodo de muestreo en segundos. Los vectores filas numz y denz especifican los coeficientes del numerador y del denominador de $G(z)$ (en potencias decrecientes de z). Así

$$G(z) = \frac{\text{numz}}{\text{denz}}$$

Para más detalles véanse los Ejemplos 6-31 y 6-32.

EJEMPLO 6-31

Suponiendo que el periodo de muestreo T es de 0.1 seg, transformar

$$G(s) = \frac{10}{s + 10}$$

en $G(z)$.

El programa de MATLAB 6-47 genera $G(z)$, la transformada z de $G(s)$ precedida por el retenedor de orden cero. Así, obtenemos

$$G(z) = \frac{0{,}6321}{z - 0{,}3679}$$

Programa en MATLAB 6-47

```
%-------Transformar G(s) a G(z)-------

%*****Introduzca el numerador y el denominador de G(s)*****

num=[0 10];
den=[1 10];

%*****El numerador (numz) y el denominador (denz) de G(z) se pueden obtener
introduciendo
%las siguientes órdenes*****

[A,B,C,D]=tf2ss(num,den);
[G,H]=c2d(A,B,0.1);
[numz,denz]=ss2tf(G,H,C,D)

numz =

         0    0.6321


denz =

    1.0000   -0.3679
```

que es correcto. Las salidas intermedias, tales como las matrices A, B, C, D, G y H se muestran a continuación:

```
num=[0 10];
den=[1 10];
[A,B,C,D]=tf2ss(num,den)

A =

   -10

B =

     1

C =

    10

D =

     0

[G,H]=c2d(A,B,0.1)

G =

    0.3679

H =

    0.0632
```

EJEMPLO 6-32

Suponiendo que el periodo de muestreo T es de 0.1 seg, transformar

$$G(s) = \frac{1}{s(s+1)}$$

en $G(z)$.

El programa de MATLAB 6-48 genera $G(z)$. Los vectores filas numz y denz dan los coeficientes de los polinomios del numerador y del denominador de $G(z)$ como sigue:

$$G(z) = \frac{0{,}01873z + 0{,}01752}{z^2 - 1{,}8187z + 0{,}8187}$$

Programa en MATLAB 6-48

```
%-------Transformar G(s) a G(z)-------

%*****Introduzca el numerador y el denominador de G(s)*****

num=[0 0 1];
den=[1 1 0];

%*****El numerador (numz) y el denominador (denz) de G(z) se pueden obtener
%introduciendo las siguientes órdenes*****

[A,B,C,D]=tf2ss(num,den);
[G,H]=c2d(A,B,0.2);
[numz,denz]=ss2tf(G,H,C,D)

numz =

         0    0.0187    0.0175


denz =

    1.0000   -1.8187    0.8187


%*****Para aumentar la precision utilizamos formato largo*****

format long
[numz,denz]=ss2tf(G,H,C,D)

numz =

                  0   0.01873075307798   0.01752309630642


denz =

   1.00000000000000  -1.81873075307798   0.81873075307798

%*****Así se obtuvo G(z) como sigue
%G(z)=(0.01873z+0.01752)/(z^2-1.8187z+0.8187)*****

%*****Restaure el formato normal*****

format short
```

Transformación bilineal

La transformación bilineal es una transformación matemática de variables. La orden bilinear transforma $G(w)$ en $G(z)$ tal que

$$G(z) = G(w)\Big|_{w = 2f_s \frac{z-1}{z+1}}$$

donde $f_s = 1/T$. (T es el periodo de muestreo en segundos). Definir

$$G(w) = \frac{\text{num}}{\text{den}}$$

Entonces la orden

[numz, denz] = bilinear(num,den,fs)

donde fs = 1/T = frecuencia de muestreo en hertz, producirá

$$G(z) = \frac{\text{numz}}{\text{denz}}$$

Observe que el diseño de controladores digitales basado en el método del diagrama de Bode utiliza dos transformaciones bilineales diferentes. Una es la transformación de $G(z)$ a $G(w)$ con la transformación bilineal definida por

$$z = \frac{1 + \frac{T}{2}w}{1 - \frac{T}{2}w} \tag{6-18}$$

donde T es el periodo de muestreo empleado en el sistema de control discreto. Una vez que la función de transferencia de la planta se transforma en $G(w)$, podemos diseñar el controlador digital $G_c(w)$ en el plano w.

La otra transformación utilizada en el proceso de diseño es transformar $G_c(w)$ en $G_c(z)$. La transformación del plano w en el plano z se efectúa con la siguiente transformación:

$$w = \frac{2}{T}\frac{z-1}{z+1} = 2f_s\frac{z-1}{z+1} \tag{6-19}$$

MATLAB realiza la transformación bilineal definida por (6-19). Por tanto la orden bilinear puede utilizarse para transformar $G(w)$ en $G(z)$. Sin embargo, esta orden no se puede usar para transformar $G(z)$ en $G(w)$.

Sin embargo es posible utilizar la orden bilinear para transformar $G(z)$ en $G(w)$ con ciertas modificaciones en las sentencias de la orden. Para más detalles véanse los Ejemplos 6-33 y 6-34.

EJEMPLO 6-33

Transformar

$$G(z) = \frac{0{,}6321}{z - 0{,}3679}$$

en $G(w)$, donde la transformación entre z y w está dada por

$$z = \frac{1 + \frac{T}{2}w}{1 - \frac{T}{2}w} = \frac{1 + 0{,}05w}{1 - 0{,}05w} \tag{6-20}$$

donde T = 0.1 seg.

Como la orden bilinear usa la transformación

$$z = 2f_s \frac{v-1}{v+1}$$

en primer lugar fijamos $f_s = 0.5$ así que

$$z = \frac{v-1}{v+1}$$

A continuación transformar v en w mediante la ecuación

$$v = -0{,}05w$$

De esta forma tenemos

$$z = \frac{-0{,}05w-1}{-0{,}05w+1} = \frac{-(1+0{,}05w)}{1-0{,}05w}$$

o

$$-z = \frac{1+0{,}05w}{1-0{,}05w} \tag{6-21}$$

Obsérvese que el lado derecho de (6-21) es igual al de (6-20).

Del análisis anterior, para transformar $G(z)$ en $G(w)$ utilizando la orden bilinear, podemos proceder como sigue:

1. Sustituir z por -z en $G(z)$ y definir

```
num = [0  0.6321];
den = [-1  -0.3679];
```

2. Utilizar la siguiente orden bilinear:

```
[numv,denv] = bilinear(num,den,fs)
```

donde fs = 0.5. Observe que numv y denv se pueden obtener como se muestra en la siguiente salida de MATLAB.

```
num=[0 0.6321];
den=[-1 -0.3679];
[numv,denv]=bilinear(num,den,0.5)

numv =

   -0.4621   -0.4621


denv =

    1.0000   -0.4621
```

Por tanto

$$G(v) = \frac{\text{numv}}{\text{denv}} = \frac{-0{,}4621v-0{,}4621}{v-0{,}4621}$$

3. Sustituir v = -0.5w en el numerador y denominador de $G(v)$. Obtenemos

$$G(w) = \frac{-0{,}4621(-0{,}05w) - 0{,}4621}{-0{,}05w - 0{,}4621}$$

Esta operación de sustituir v = -0.5w en el numerador y denominador de $G(v)$ puede hacerse con MATLAB de la forma siguiente. En primer lugar observe que

numerador en w = -0.4621(-0.05w) - 0.4621

Así,

```
numw = [-0.4621   -0.4621].*[-0.05   1]
```
(6-22)

Aquí la operación '.*' significa que esta multiplicación se hace elemento a elemento. Análogamente

denominador en w = -0.05w - 0.4621

de donde obtenemos

```
denw = [1   -0.4621].*[-0.05   1]
```
(6-23)

Las ecuaciones (6-22) y (6-23) se pueden escribir como

```
numw = [numv].*[-0.05   1]
denw = [denv].*[-0.05   1]
```

Introduciendo estas órdenes en el computador, obtenemos $G(w)$. Véase la siguiente salida de MATLAB.

```
numw=[numv].*[-0.05 1], denw=[denv].*[-0.05 1]

numw =

    0.0231   -0.4621


denw =

   -0.0500   -0.4621
```

El primer término en numw se imprime como 0.0231. Para comprobar la precisión de este número impreso, lo imprimimos con *format long*.

```
format long
numw=[numv].*[-0.05 1], denw=[denv].*[-0.05 1]

numw =

   0.02310475911982  -0.46209518239637


denw =

  -0.05000000000000  -0.46209518239637
```

De esta salida *format long*, obtenemos

$$G(w) = \frac{0{,}023105w - 0{,}462095}{-0{,}05w - 0{,}462095} \tag{6-24}$$

Aunque esta expresión es correcta, es conveniente si el coeficiente del término de mayor grado en w del denominador es la unidad, o

$$G(w) = \frac{-0{,}4621w + 9{,}2419}{w + 9{,}2419}$$

Esto se puede hacer multiplicando -20 por el numerador y el denominador de $G(w)$ dada por (6-24). Por tanto, si introducimos el programa

```
numw = [numv].*[1   -20]
denw = [denv].*[1   -20]
```

entonces la salida del computador tendrá la forma deseada, como se ve de la siguiente salida de MATLAB.

```
numw=[numv].*[1 -20], denw=[denv].*[1 -20]

numw =

   -0.4621    9.2419


denw =

    1.0000    9.2419
```

El programa de MATALB 6-49 muestra una serie de órdenes para transformar $G(z)$ en $G(w)$.

Programa en MATLAB 6-49

```
%-------Transformar G(z) a G(w)-------

%*****Primero sustituya -z por z en la funcion de transferencia G(z) e
%introduzca el numerador y el denominador*****

num=[0 0.6321];
den=[-1 -0.3679];

%*****Itroduzca la orden 'bilinear' con fs=0.5*****

[numv,denv]=bilinear(num,den,0.5);

%*****Transforme G(v) a G(w) donde v y w están relacionadas por v=-0.05w.
%Esto se puede realizar introduciendo las siguientes órdenes*****

numw=[numv].*[1 -20],denw=[denv].*[1 -20]


%*****Así se obtiene G(w) como sigue
%G(w)=(-0.4621w+9.2419)/(w+9.2419)*****
```

EJEMPLO 6-34

Transformar

$$G(z) = \frac{0{,}01873z + 0{,}01752}{z^2 - 1{,}8187z + 0{,}8187}$$

en $G(w)$, donde la transformación entre z y w viene dada por

$$z = \frac{1 + \frac{T}{2}w}{1 - \frac{T}{2}w} = \frac{1 + 0{,}1w}{1 - 0{,}1w}$$

donde T = 0.2 seg.

Para obtener $G(w)$, podemos proceder como sigue:

1. Sustituir z por -z en $G(z)$ y definir

```
num = [0   -0.01873   0.01752]
den = [1   1.8187   0.8187]
```

2. Utilizar la siguiente orden bilinear:

```
[numv,denv] = bilinear(num,den,0.5)
```

La salida de MATLAB es como sigue:

```
[numv,denv]=bilinear(num,den,0.5)

numv =

   -0.0003    0.0096    0.0100


denv =

    1.0000   -0.0997    0.0000
```

3. Convertir $G(\nu)$ en $G(w)$, observar que ν y w están relacionadas por ν = -0.1w. Como

numerador en $v = -0.0003v^2 + 0.0096v + 0.0100$

tenemos

numerador en $w = -0.0003(-0.1w)^2 + 0.0096(-0.1w) + 0.0100$

Por tanto

```
numw = [-0.0003   0.0096   0.0100].*[(-0.1)^2   -0.1   1]
```

Análogamente

denominador en $v = v^2 - 0.0997v$

y

denominador en $w = (-0.1w)^2 - 0.0997(-0.1w)$

De donde se deduce que

```
denw = [1   -0.0997   0].*[(-0.1)^2   -0.1   1]
```

Para hacer que el coeficiente de la potencia más elevada del denominador sea la unidad, multiplicar 100 por

```
[(-0.1)^2   -0.1   1]
```

y escribir numw y denw como sigue

```
numw = [numv].*[1   -10   100];
denw = [dev].*[1   -10   100];
```

Basado en los comentarios anteriores, podemos introducir el programa de MATLAB 6-50 para obtener $G(w)$. La función de transferencia $G(w)$ obtenida por el uso de este programa se puede escribir como sigue:

$$G(w) = \frac{-0{,}000333w^2 - 0{,}09633w + 0{,}9966}{w^2 + 0{,}9969w}$$

$$\cong \frac{\left(1 + \frac{w}{300}\right)\left(1 - \frac{w}{10}\right)}{w(w+1)} \tag{6-25}$$

Observe que en este ejemplo $G(z)$ se obtuvo a partir de la siguiente $G(s)$:

$$G(s) = \frac{1}{s(s+1)} \tag{6-26}$$

(Véase el Ejemplo 6-32). Comparando las ecuaciones (6-25) y (6-26), las funciones de transferencia $G(w)$ y $G(s)$ son similares ya que ambas tienen la misma ganancia de continua y tienen los polos en las mismas localizaciones.

La diferencia aparece en las localizaciones de los ceros. $G(s)$ no tiene ceros finitos, mientras que $G(w)$ tiene dos ceros, uno en w = -300 (que está localizado muy lejos del origen y por tanto su influencia sobre la respuesta transitoria es insignificante) y el otro en el semiplano derecho en w = 10.

Programa en MATLAB 6-50

```
%-------Transformar G(z) a G(w)-------

%*****Primero sustituya -z por z en la funcion de transferencia G(z)
% e introduzca el numerador y el denominador*****

num=[0 -0.01873 0.01752];
den=[1 1.8187 0.8187];

%*****Introduzca la orden 'bilinear' con fs=0.5*****

[numv,denv]=bilinear(num,den,0.5);

%*****Transforme G(v) a G(w) donde v y w están relacionadas por v=-0.1w.
%Esto se puede realizar introduciendo las siguientes órdenes*****

numw=[numv].*[1 -10 100],denw=[denv].*[1 -10 100]


%*****Para aumentar la precision, se puede utilizar el formato largo*****

format long
numw=[numv].*[1 -10 100],denw=[denv].*[1 -10 100]


%*****Así se obtiene G(w) como sigue
%G(w)=(-0.000333w^2-0.09633w+0.9966)/(w^2+0.9969w)*****

%*****restaure el formato normal*****

format short
```

Por tanto, $G(w)$ es una función de transferencia de fase no mínima. Las diferencias en el diagrama de Bode de $G(s)$ y $G(w)$ se pueden observar en la Figura 6-74, que se obtuvo mediante el programa de MATLAB 6-51.

Observe que las curvas de magnitud son aproximadamente iguales para $G(s)$ y $G(w)$ en el rango de frecuencias $0 < w < 6$ rad/seg. La separación de las curvas de magnitud ocurre alrededor de $w = 6$ rad/seg. La magnitud $|G(s)|$ tiende a $-\infty$ dB cuando la frecuencia aumenta, mientras que la magnitud de $|G(w)|$ tiende a un valor constante de -69.5 dB cuando la frecuencia aumenta. Las curvas de fase son las mismas para $0 < w < 0.2$ y $10^4 < w < \infty$. Para $0.2 < w < 10^4$, aparece una diferencia significativa porque uno es un sistema de fase mínima y el otro es de fase no mínima. [La contribución de ángulo del cero en el semiplano derecho w es negativa (retardo de fase)].

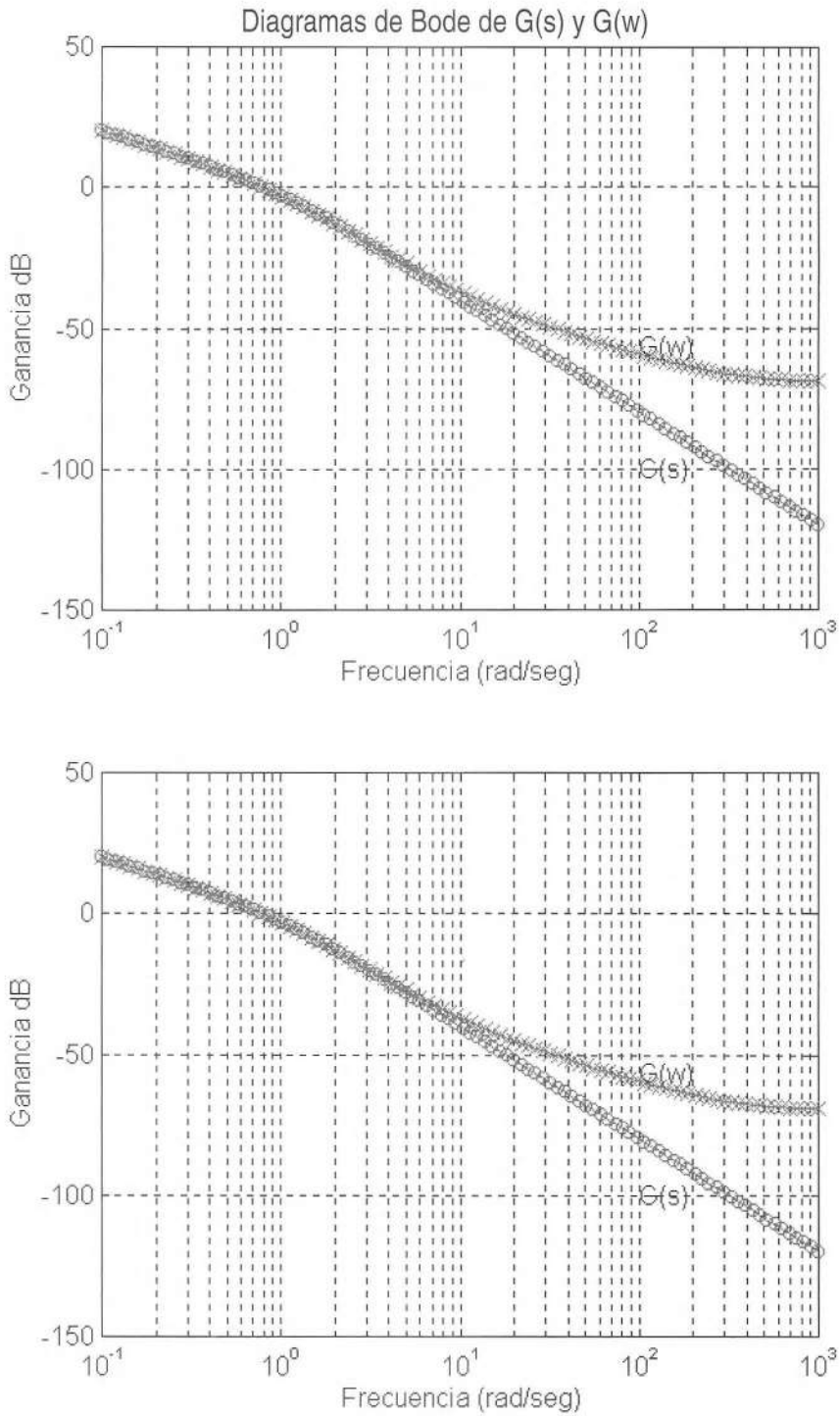

Diagramas de Bode de G(s) y G(w)
Ganancia dB
Frecuencia (rad/seg)
G(w)
G(s)
50
0
-50
-100
-150
10^-1
10^0
10^1
10^2
10^3

Figura 6-74

Programa en MATLAB 6-51

```
%-------Diagramas de Bode-------

%*****Se dibujarán los diagramas de Bode de G(s) y G(w)*****

nums=[0 0 1]; dens=[1 1 0];
numw=[-0.000333 -0.09633 0.9966]; denw=[1 0.9969 0];
w=logspace(-1,3,100);
[mags,fases,w]=bode(nums,dens,w);
magsdB=20*log10(mags);
[magw,fasew,w]=bode(numw,denw,w);
magwdB=20*log10(magw);

%*****Primero se dibujaran las curvas de magnitud*****

semilogx(w,magsdB,'o',w,magsdB,'-')
hold

%Current plot held

semilogx(w,magwdB,'x',w,magwdB,'-')
grid, title('Diagramas de Bode de G(s) y G(w)')
xlabel('Frecuencia (rad/seg)')
ylabel('Ganancia dB')
text(100,-100,'G(s)')
text(100,-57,'G(w)')
hold

%Current plot released

%*****A continuacion se dibujaran las curvas de fase*****

%*****Se elije como rango de fase de -300 grados a -50 grados. Esto se puede
%realizar dibujando lineas no visibles en -290 grados y -60 grados*****

pmax=-60*ones(1,100);
pmin=-290*ones(1,100);
semilogx(w,fases,'o',w,fases,'-',w,pmax,'--i',w,pmin,':i')
hold

%Current plot held

semilogx(w,fasew,'x',w,fasew,'-')
grid
xlabel('Frecuencia (rad/seg)')
ylabel('Fase grados')
text(10,-170,'G(s)')
text(10,-245,'G(w)')

%*****Elimine los graficos retenidos*****

hold

%Current plot released
```

EJEMPLO 6-35

Sea el sistema de control digital que se muestra en la Figura 6-75. Diseñar un controlador digital en el plano w tal que el margen de fase sea de 50°, el margen de ganancia de 10 dB y la constante de error de velocidad estática K_v igual a 2 seg^{-1}. Suponer que el periodo de muestreo T es de 0.2 seg.

La función de transferencia discreta $G(z)$ de la planta es

$$G(z) = Z\left[\frac{1-e^{-0,2s}}{s}\frac{K}{s(s+1)}\right]$$

$$= (1-z^{-1})Z\left[\frac{K}{s^2(s+1)}\right]$$

$$= \frac{K(0,01873z+0,01752)}{z^2-1,8187z+0,8187}$$

Este resultado (cuando K = 1) se obtuvo en el Ejemplo 6-32.

Para aplicar el método del diagrama de Bode al diseño, necesitamos transformar $G(z)$ en $G(w)$. La transformación de z en w viene dada por

$$z = \frac{1+\frac{T}{2}w}{1-\frac{T}{2}w} = \frac{1+0,1w}{1-0,1w}$$

De donde $G(w)$ viene dada por

$$G(w) = \left.\frac{K(0,01873z+0,01752)}{z^2-1,8187z+0,8187}\right|_{z=\frac{1+0,1w}{1-0,1w}}$$

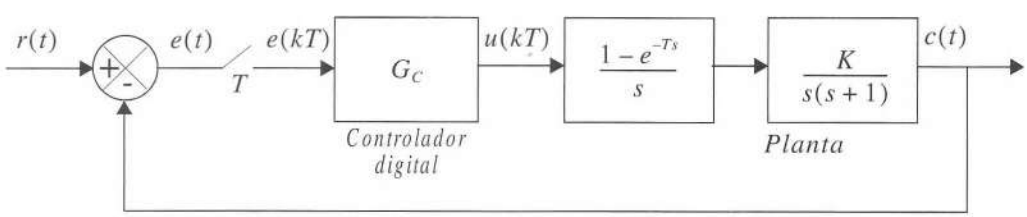


Figura 6-75

Esta transformación de $G(s)$ a $G(s)$ cuando K = 1 fue realizada por MATLAB en el Ejemplo 6-34. Al referirse al resultado obtenido allí, tenemos

$$G(w) = \frac{K(-0,000333w^2-0,09633w+0,9966)}{w^2+0,9969w}$$

$$\equiv K\frac{\left(1+\frac{w}{300}\right)\left(1-\frac{w}{10}\right)}{w(w+1)}$$

La forma factorizada de $G(w)$ nos dice dónde están localizadas las frecuencias esquinas. Observe que un cero se localiza en el semiplano derecho w (w = 10). $G(w)$ es por tanto una función de transferencia de fase no mínima.

Ahora asumimos que la función de transferencia del controlador digital $G_c(w)$ tiene ganancia unidad en el rango de frecuencias bajas y tiene la forma siguiente:

$$G_c(w) = \frac{1 + \tau w}{1 + \alpha\tau w}$$

(Observe que ésta es una de las formas más simples para la función de transferencia del controlador digital. Se pueden suponer también otras formas). La función de transferencia en lazo abierto es

$$G_c(w)G(w) = \frac{1 + \tau w}{1 + \alpha\tau w} \frac{K(-0{,}000333w^2 - 0{,}09633w + 0{,}9966)}{w^2 + 0{,}9969w}$$

La constante de error de velocidad estática K_v se especifica como 2 seg^{-1}. Por tanto

$$K_v = \lim_{w \to 0} wG_c(w)G(w) \cong K = 2$$

La ganancia K se determina así como 2

$$K = 2$$

A continuación, fijando $K = 2$, representaremos el diagrama de Bode de $G(w)$:

$$G(w) = \frac{-0{,}000666w^2 - 0{,}19266w + 1{,}9932}{w^2 + 0{,}9969w}$$

El programa de MATLAB 6-52 se puede utilizar para producir el diagrama de Bode de $G(w)$ con $K = 2$.

La Figura 6-76 muestra el diagrama de Bode de $G(w)$ con $K = 2$. El margen de fase se puede leer del diagrama como 30° y el margen de ganancia como 14.5 dB.

La especificación dada requiere, ademas de $K_v = 2$, el margen de fase de 50° y un margen de ganancia de al menos 10 dB. Diseñemos un controlador digital para satisfacer estas especificaciones. Emplearemos la técnica de diseño convencional tal como se da a continuación

Programa en MATLAB 6-52

```
num=[-0.000666 -0.19266 1.9932];
den=[1 0.9969 0];
w=logspace(-1,3,100);
bode(num,den,w)
title('Diagrama de Bode de la planta G(w) con K=2')
```

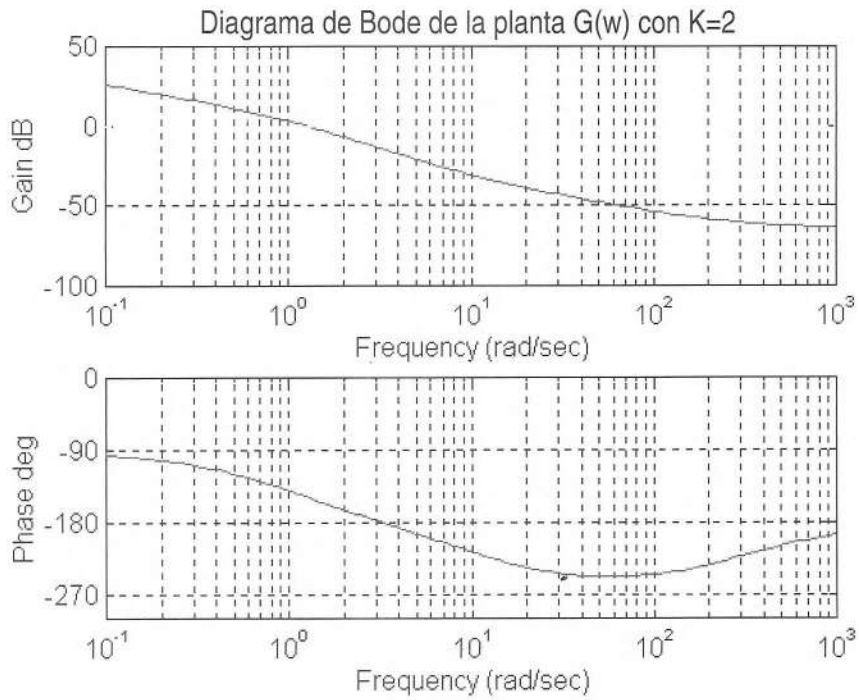


Figura 6-76

Diseño del compensador de adelanto: Como la especificación busca un margen de fase de 50°, el ángulo de adelanto de fase adicional que es necesario para satisfacer este requerimiento es de 20°. Para conseguir un margen de fase de 50° sin disminuir el valor de K, el compensador de adelanto debe contribuir con el ángulo de fase requerido.

Hay que tener en cuenta que la inclusión de un compensador de adelanto modifica la curva de magnitud del diagrama de Bode y que la frecuencia de la ganancia de cruce se desplazará hacia la derecha. Considerando este desplazamiento podemos asumir que ϕ_m, el máximo adelanto de fase que se necesita es aproximadamente de 28°. (Esto significa que se han añadido 8° para compensar el desplazamiento en la fracuencia de la ganancia de cruce). Como

$$\operatorname{sen}\phi_m = \frac{1-\alpha}{1+\alpha}$$

$\phi_m = 28°$ corresponde a $\alpha = 0.361$.

Una vez que se ha determinado el factor de atenuación α teniendo en cuenta el ángulo de adelanto de fase que se necesita, el siguiente paso es determinar las frecuencias esquinas $w = 1/\tau$ y $w = 1/(\alpha\tau)$ del compensador de adelanto. Para hacerlo, en primer lugar observamos que el ángulo máximo de adelanto de fase ϕ_m ocurre en la media geométrica de las dos frecuencias esquinas, o $w = 1/(\sqrt{\alpha}\tau)$. La cantidad de la modificación en la curva de magnitud en $w = 1/(\sqrt{\alpha}\tau)$ debido a la inclusión del término $(1 + \tau\omega)/(1 + \alpha\tau\omega)$ es

$$\left|\frac{1+\tau w}{1+\alpha\tau w}\right|_{w=\frac{1}{\tau\sqrt{\alpha}}} = \frac{1}{\sqrt{\alpha}}$$

Después calculamos la frecuencia donde la magnitud del sistema sin compensar es igual a $-20 \log (1/\sqrt{\alpha})$. Observe que

$$-20 \log \frac{1}{\sqrt{0{,}361}} = -20 \log 1{,}6643 = -4{,}425 \text{ dB}$$

Para encontrar el punto de frecuencia donde la magnitud es -4.425 dB, podemos representar el diagrama de Bode en el rango de frecuencias $1 \leq w \leq 10$ rad/seg. Véase el programa de MATLAB 6-53 y la gráfica resultante que se muestra en la Figura 6-77. De este diagrama, encontramos que en w = 1.7 rad/seg, la magnitud se hace aproximadamente -4.4 dB.

Seleccionamos esta frecuencia para que sea la nueva frecuencia da la ganacia de cruce w_c. Si se tiene en cuenta que esta frecuencia corresponde a $w = 1/(\sqrt{\alpha}\tau)$, o

$$\omega_c = \frac{1}{\tau\sqrt{\alpha}} = 1{,}7$$

obtenemos

$$\tau = \frac{1}{1{,}7\sqrt{\alpha}} = 0{,}9790$$

y

$$\alpha\tau = 0{,}3534$$

Programa en MATLAB 6-53

```
num=[-0.000666 -0.19266 1.9932];
den=[1 0.9969 0];
w=logspace(0,1,100);
bode(num,den,w)
title('Diagrama de Bode de la planta G(w) con K=2')
```

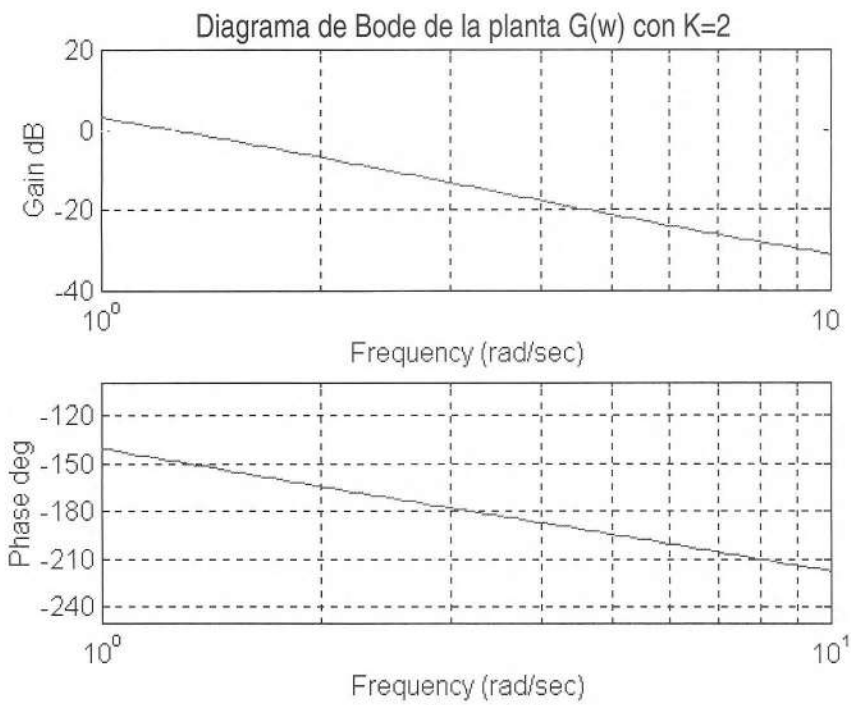


Figura 6-77

El compensador de adelanto así determinado es

$$G_c(w) = \frac{1 + \tau w}{1 + \alpha\tau w} = \frac{1 + 0{,}9790w}{1 + 0{,}3534w} \tag{6-27}$$

En la Figura 6-78 se muestra el el diagrama de Bode de este compensador de adelanto. Se obtiene mediante el uso del Programa de MATLAB 6-54.

Para representar los diagramas de Bode de $G(w)$, $G_c(w)$ y $G_c(w)G(w)$ se puede utilizar el Programa de MATLAB 6-55. Las tres curvas de magnitud se representan en una gráfica y las tres de ángulo de fase en otra. En la Figura 6-79 se muestran estas gráficas.

De las curvas de magnitud y de ángulo de fase de $G_c(w)G(w)$, encontramos que el margen de fase es aproximadamente de 50° y el de ganancia de 14 dB.

Como la ganancia K se ajusta para satisfacer el requerimiento sobre K_v, el sistema compensado satisface todos los requisitos. Por tanto el controlador digital $G_c(w)$ tal como se ha diseñado es satisfactorio. [Si $G_c(w)G(w)$ no cumple todas las especificaciones, sería necesario desplazar las frecuencias esquinas de $G_c(w)$].

Observe que la solución a este problema de diseño no es única. Muchas otras $G_c(w)$'s satisfacerán todos los requisitos.

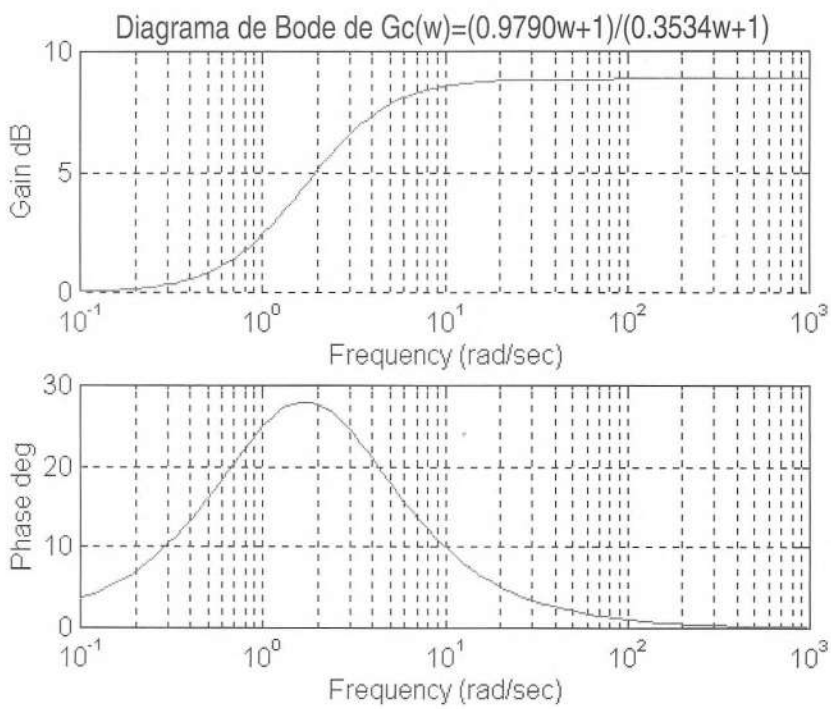


Figura 6-78

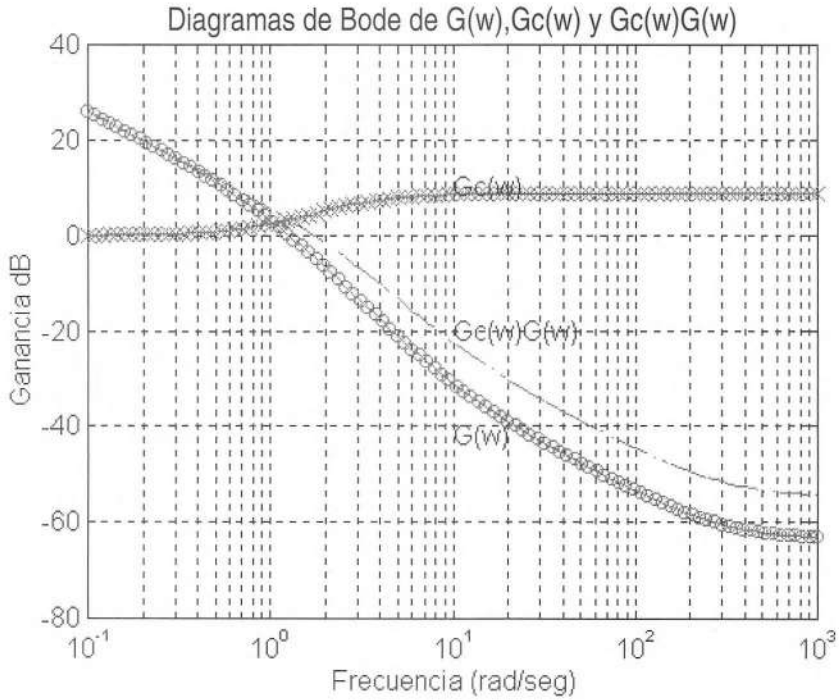
Programa en MATLAB 6-54

```
numc=[0.9790 1];
denc=[0.3534 1];
w=logspace(-1,3,100);
bode(numc,denc,w)
title('Diagrama de Bode de Gc(w)=(0.9790w+1)/(0.3534w+1)')
```

Programa en MATLAB 6-55

```
%-------Diagramas de Bode-------

%*****En este programa se dibujan los diagramas de Bode de
%G(s), G(w) y Gc(w)G(w)*****

num=[-0.000666 -0.19266 1.9932]; den=[1 0.9969 0];
numc=[0.9790 1]; denc=[0.3534 1];
w=logspace(-1,3,100);
[mag,fase,w]=bode(num,den,w);
magdB=20*log10(mag);
[magc,fasec,w]=bode(numc,denc,w);
magcdB=20*log10(magc);

%*****Definamos la magnitud de Gc(w)G(w)en decibelios como mgdB.
%mgdB viene dado por mgdB=magdB+magcdB*****

mgdB=magdB+magcdB;

%*****Primero se dibujaran las curvas de magnitud*****

semilogx(w,magdB,'o',w,magdB,'-',w,magcdB,'x',w,magcdB,'-',w,mgdB,'--')
grid
title('Diagramas de Bode de G(w),Gc(w) y Gc(w)G(w)')
xlabel('Frecuencia (rad/seg)')
ylabel('Ganancia dB')
text(10,-42,'G(w)')
text(10,10,'Gc(w)')
text(10,-20,'Gc(w)G(w)')

%*****A continuacion se dibujarán las curvas de fase*****

%*****Definamos la fase de Gc(w)G(w)en decibelios como fa.
%fa viene dado por fa=fase+fasec*****

fa=fase+fasec;

%*****Se elije como rango de fase de -300 grados a +50 grados. Esto se puede
%realizar dibujando lineas no visibles en -290 grados y +40 grados. Tambien
%podemos dibujar una linea en -180 grados*****

pmax=40*ones(1,100);
pmin=-290*ones(1,100);
ph180=-180*ones(1,100);

semilogx(w,fases,'o',w,fases,'-',w,pmax,'--i',w,pmin,':i')
hold

%*****Introduzca la siguiente orden de dibujo*****

semilogx(w,fase,'o',w,fase,'-',w,fasec,'x',w,fasec,'-',...
   w,fa,'--',w,pmax,'-i',w,pmin,'-i',w,ph180,'-')
grid
xlabel('Frecuencia (rad/seg)')
```

```
ylabel('Fase grados')
text(1,-177,'G(w)')
text(1,4,'Gc(w)')
text(1,-115,'Gc(w)G(w)')
```



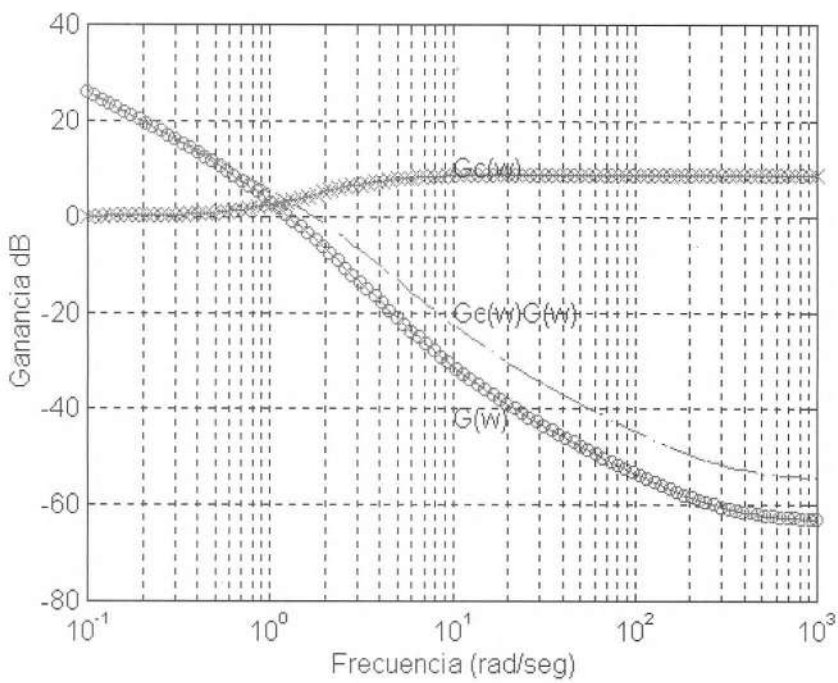


Figura 6-79

La función de transferencia del controlador dada por (6-27) será ahora transformada al plano z mediante la utilización de la siguiente transformación bilineal:

$$w = \frac{2}{T}\frac{z-1}{z+1} = 10\frac{z-1}{z+1}$$

El cálculo

$$G_c(z) = G_c(w)\Big|_{w=10\frac{z-1}{z+1}} = \frac{1+0{,}9790w}{1+0{,}3534w}\Bigg|_{w=10\frac{z-1}{z+1}}$$

se puede manejar fácilmente con la siguiente orden bilinear:

[numcz,dencz] = bilinear(numc,denc,fs)

donde fs en este caso es

$$\text{fs} = \frac{1}{T} = \frac{1}{0{,}2} = 5$$

Se puede utilizar el Programa de MATLAB 6-56 para transformar $G_c(w)$ en $G_c(z)$. La función de transferencia discreta $G_c(z)$ del controlador viene ahora dada por

$$G_c(z) = \frac{2{,}3798z - 1{,}9387}{z - 0{,}5589}$$

La función de transferencia en lazo abierto del sistema compensado es:

$$G_c(z)G(z) = \frac{2{,}3798z - 1{,}9387}{z - 0{,}5589}\,\frac{0{,}03746z + 0{,}03504}{z^2 - 1{,}8187z + 0{,}8187}$$

Programa en MATLAB 6-56

```
%-------Transformacion de Gc(w) a Gc(z)-------

numc=[0.9790 1];
denc=[0.3534 1];
[numcz,dencz]=bilinear(numc,denc,5)

numcz =

    2.3798   -1.9387


dencz =

    1.0000   -0.5589
```

La multiplicación de los dos polinomios del numerador y del denominador de esta función se puede hacer fácilmente utilizando la orden *conv* de MATLAB, tal como sigue

```
a=[2.3798 -1.9837];
b=[0.03746 0.03504];
p=conv(a,b)

p =

    0.0891    0.0091   -0.0695

c=[1 -0.5589];
d=[1 -1.8187 0.8187];
q=conv(c,d)

q =

    1.0000   -2.3776    1.8352   -0.4576
```

Por tanto

$$G_c(z)G(z) = \frac{0{,}0891z^2 + 0{,}0108z - 0{,}0679}{z^3 - 2{,}3776z^2 + 1{,}8352z - 0{,}4576}$$

La función de transferencia en lazo cerrado del sistema diseñado es:

$$\frac{C(z)}{R(z)} = \frac{0{,}0891z^2 + 0{,}0108z - 0{,}0679}{z^3 - 2{,}2885z^2 + 1{,}8460z - 0{,}5255}$$

Los ceros y los polos de la función de transferencia en lazo cerrado se pueden obtener tal como se muestra en la siguiente salida de MATLAB.

```
num=[0 0.0891 0.0108 -0.0679];
den=[1 -2.2885 1.8460 -0.5255];
roots(num)

ans =

   -0.9357
    0.8145

roots(den)

ans =

   0.8126
   0.7379 + 0.3196i
   0.7379 - 0.3196i
```

Observe que la función de transferencia discreta en lazo cerrado tiene dos ceros localizados en z =-0.9357 y z = 0.8145. El cero en z = 0.8145 casi se cancela con el polo en lazo cerrado en z = 0.8126. El efecto del otro cero en z = -0.9357 sobre el transitorio y la respuesta en frecuencia es muy pequeño, ya que se encuentra localizado sobre el eje real negativo del plano z entre 0 y -1 y está próximo al punto z = -1.

Para comprobar las características de la respuesta transitoria, obtendremos la respuesta a un salto unitario del sistema diseñado utilizando el programa de MATLAB 6-57. En la Figura 6-80 se muestra la curva de respuesta resultante. La curva de respuesta a un salto unitario exhibe una sobreelongación máxima del 20% y un tiempo de asentamiento de aproximadamente 4 seg.

Observe que para este sistema el periodo de muestreo T se escoge de 0.2 seg. De la Figura 6-80 se puede ver que el número de muestras por ciclo de oscilación sinusoidal es de 14. Así pues la frecuencia de muestreo es 14 veces la frecuencia natural amortiguada y el periodo de muestreo es satisfactorio bajo circunstancias normales. Si el sistema no tuviese ni controlador digital ni retenedor de orden cero, pero la ganancia K se ajustase simplemente a 2, el sistema en lazo cerrado tendría la función de transferencia

$$\frac{C(s)}{R(s)} = \frac{2}{s^2 + s + 2}$$

y la sobreelongación máxima habría sido del 30% y el tiempo de asentamiento de 8 seg. Así el sistema diseñado muestra una mejora considerable de la respuesta.

Programa en MATLAB 6-57

```
%-------Respuesta a un salto unitario del sistema diseòado-------

num=[0 0.0891 0.0108 -0.0679];
den=[1 -2.2885 1.8460 -0.5255];
x=ones(1,41);
v=[0 40 0 1.6];
axis(v);
k=0:40;
y=filter(num,den,x);
plot(k,y,'o')
grid
title('Respuesta a un salto unitario del sistema diseòado')
xlabel('k (Periodo de muestreo T=0.2 seg)')
ylabel('Salida c(k)')
```

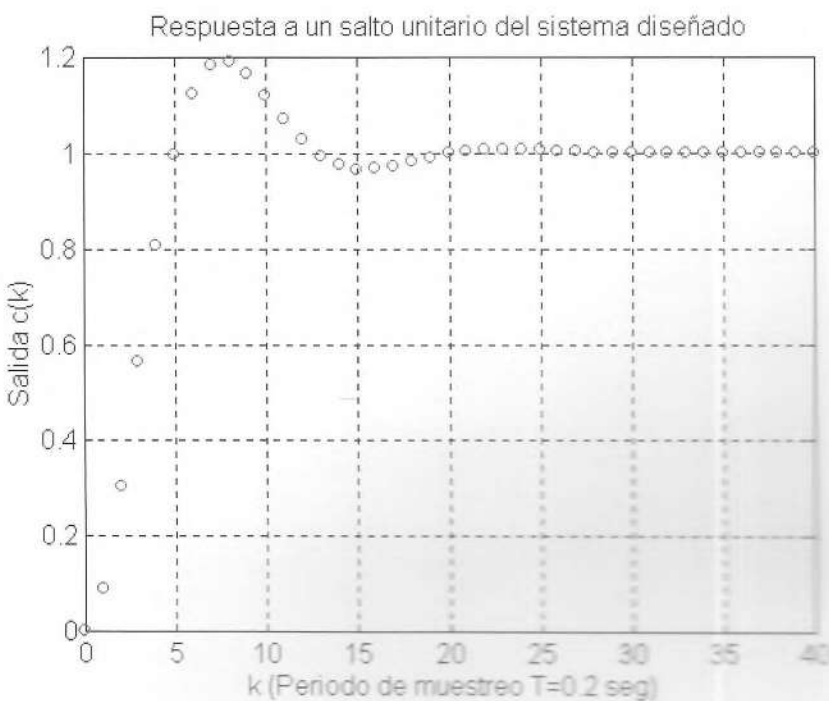


Figura 6-80

Comentarios

La ventaja de la transformación w es que el método de respuesta en frecuencia convencional utilizando el diagrama de Bode se puede emplear para el diseño de sistemas de control en tiempo discreto.

Recuerde que la transformación w puede generar uno o más ceros en el semiplano derecho en $G(w)$. Si existen uno o más ceros en el semiplano derecho, entonces $G(w)$ es una función de transferencia de fase no mínima. Porque los ceros en el semiplano derecho se generan por la operación de muestreo y retención, la localización de estos ceros depende del periodo de muestreo T. Los efectos de estos ceros en el semiplano derecho sobre la respuesta se hacen más pequeños cuando disminuye el periodo de muestreo T. (Un periodo de muestreo más pequeño significa una frecuencia de muestreo más elevada).